# Secure and Efficient Peer2Peer Authentication–Attestation Protocol for UAV Networks

Gaurang Bansal, *Member, IEEE*, and Biplab Sikdar, *Senior Member, IEEE*

***Abstract*—The accelerating deployment of unmanned aerial vehicles (UAVs) is accompanied by escalating security concerns, especially with regard to communication protocols. Traditional cryptographic mechanisms, while functional, fall short in computational efficiency and low-latency requirements that are critical for UAV networks. Addressing these challenges, this article introduces a novel hardware-secured authentication and attestation mechanism tailored for UAV-to-UAV data exchange. The mechanism is designed to scale efficiently with UAV swarms and withstand rigorous post-deployment verifications. Our research contributions are multifaceted, comprising: 1) a feasibility and security validation of the proposed protocol via Mao–Boyd logic, providing a robust theoretical foundation; 2) empirical results that confirm the protocol's superior performance over contemporary solutions in both speed and security; and 3) a comprehensive security and performance analysis to ensure the protocol's resilience against potential vulnerabilities. Thus, this article presents a balanced and effective approach to secure UAV communications, satisfying both computational and security demands.**

***Index Terms*—Attestation, authentication, distributed networks, peer networking, unmanned aerial vehicles (UAVs).**

## I. Introduction

IN RECENT years, there has been rise of unmanned aerial vehicle (UAV)-based applications. The impetus behind this surge lies not just in technological advancements, but also in the diverse use-cases that UAVs present, ranging from agriculture and disaster management to logistics and transportation. Nonetheless, the proliferation of UAVs brings forth a set of compelling security challenges. In particular, UAV communication systems are susceptible to a wide spectrum of security threats, including but not limited to, replay attacks and intruder attacks, as evidenced by existing literature [1], [2].

Manuscript received 15 January 2024; revised 17 March 2024, 13 April 2024, and 18 May 2024; accepted 17 June 2024. Date of publication 1 July 2024; date of current version 6 December 2024. This work was supported in part by the National Research Foundation, Singapore, and Infocomm Media Development Authority under its Future Communications Research Development Programme, under Grant FCP-NUS-RG-2022-019. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and Infocomm Media Development Authority. *(Corresponding author: Gaurang Bansal.)*

The authors are with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 119077 (e-mail: e0622339@u.nus.edu; bsikdar@u.nus.edu).

Recent UAV-related security incidents demonstrate the critical vulnerabilities and evolving risks posed by UAVs [3]. Examples include the following.

1) *Corporate Network Breach:* Attackers utilized drones equipped with Wi-Fi spoofing devices to penetrate a U.S. financial firm's network, showcasing the sophistication of such cyberattacks.
2) *Airport Security Disruptions:* Unauthorized drone activity at Reagan Washington National Airport led to a 13-min flight suspension, impacting 90 flights and causing extensive operational disruptions.
3) *Smuggling Operations:* Drones have been used for illicit activities, such as smuggling 11 handguns from the USA to Canada, highlighting their use in criminal enterprises.

Two salient procedures in ensuring the robustness of UAV communications are authentication and attestation. Authentication serves as the gateway mechanism by which participating entities within a communication network validate each other's credentials. In the specific context of UAVs, this process gains even more pertinence given that authenticated channels between UAVs and base stations (BSs) are integral in fortifying the UAV network against adversarial intrusions [4], [5].

In parallel, attestation procedures serve as the cornerstone for validating the integrity of hardware and software components, particularly the memory chip and its resident firmware. Attestation allows for the verification of whether the firmware on a UAV has been illicitly altered, thereby serving as a bulwark against remote tampering attempts that could compromise the UAV's operational integrity [6]. This process typically involves a bidirectional communication protocol between the UAV device (the prover) and the BS (the verifier), which ensures that both parties mutually affirm the integrity of the communication [7], [8].

The authentication and attestation protocols for UAV swarms must fulfill the following requirements to ensure secure and efficient operations.

1) *Low Latency:* Protocols must ensure minimal response time to maintain real-time control and coordination among the UAVs in the swarm.
2) *Resource Efficiency:* Given the limited computational and battery resources in UAVs, protocols should consume minimal resources to maximize operational efficiency and duration.
3) *Scalability:* The system must efficiently scale as the number of UAVs in the swarm increases, without degradation in performance.

4) *Resilience to Compromise:* Protocols should be designed to limit the impact of a compromised UAV on the rest of the swarm, preventing it from affecting the operations or security of other UAVs.

In this article, we introduce an advanced hardware-secured protocol specifically engineered for authentication and attestation in UAV-to-UAV data exchange environments. This protocol is optimized for use in complex UAV fleet deployments where rigorous post-deployment verification and testing are mandatory. The protocol features scalability, low-latency, and enhanced security measures, making it highly adaptable for heterogeneous UAV swarm deployments. The key contributions of this article can be outlined as follows:

1) *Rigorous Theoretical Validation:* The manuscript employs Mao–Boyd logic as the underlying theoretical framework to rigorously validate both the feasibility and the security attributes of the proposed protocol. This analytical substantiation lends robust theoretical support, endorsing the protocol's capabilities in mitigating varied security risks while upholding a secure communication environment within UAV networks.
2) *Benchmarking Performance Metrics:* The manuscript validates through empirical evidence that the proposed protocol mechanism offers a significant performance advantage over existing state-of-the-art solutions. It accomplishes this by yielding reduced execution times without making any concessions on the security integrity, thereby underlining its practical utility for real-world UAV deployment scenarios.
3) *Comprehensive Multifaceted Analysis:* The study incorporates a multifaceted analytical approach covering both security and performance considerations. Section V is allocated for intensive security analysis, which scrutinizes the resilience of the proposed protocol against a spectrum of potential security threats, thereby confirming its robustness.

The subsequent structure of the manuscript is strategically organized for ease of comprehension. Section II provides with a review of relevant academic literature and pinpoints existing gaps in the current body of research. Section III delineates the system architecture and adversarial models pertinent to the study. A thorough description of the proposed peer-to-peer authentication and attestation protocol is presented in Section IV. Sections V and VI are dedicated to in-depth security and performance evaluations, respectively. The manuscript concludes with key findings and future directions in Section VII.

## II. Related Works

In the realm of UAVs, swarms exhibit unique characteristics, such as heightened mobility and inherently dynamic network topologies. Moreover, these UAV systems often operate under constraints related to computational capabilities and energy reserves, underscoring the need for the implementation of lightweight yet effective security measures. Due to these specialized operational conditions, the security requirements for UAV swarms significantly differ from those traditionally applied to standard ad-hoc networks. A plethora of research efforts, exemplified by the works of Fotouhi et al. [9], Sun et al. [10], Birk et al. [11], Wazid et al. [12], and Srinivas et al. [13], have been dedicated to developing security schemes that are both computationally efficient and swiftly executable, without compromising on security efficacy. This section of this article provides a concise yet comprehensive overview of existing authentication and attestation technologies proposed to ensure secure communications within UAV networks.

A methodological approach advanced by Jiang et al. [14] employed artificial intelligence techniques alongside real-time behavioral analytics, such as positional data of UAVs, to facilitate the processes of identification and authentication. Despite its innovative approach, this methodology is limited to the authentication of individual UAV units and does not extend to swarm-level security. In terms of leveraging hardware-based security, works by Alladi et al. [1], [15] utilized physical unclonable functions (PUFs) to bolster the physical security aspects during the authentication processes. Although effective against physical attack vectors, these approaches display limitations in scalability or are applicable only to specialized two-tier UAV swarm architectures. Yahuza et al. [16] introduced a scheme known as secure lightweight proven authenticated key agreement (SLPAKA), crafted with the objective of achieving network scalability. Nonetheless, this framework is susceptible to a range of physical attacks, indicating a security vulnerability. Du et al. [17] contributed another lightweight security protocol that amalgamates elements of security game theory. While the mechanism is robust against certain attack vectors, it fails to offer comprehensive protection against a broad spectrum of potential security threats, such as replay attacks, physical attacks, etc. Alternative frameworks, represented by Khanh et al. [18] and Chen et al. [19], have been designed to furnish effective authentication solutions for UAV swarms. However, they are hampered by their computational intensiveness, making them less than ideal for real-world deployments. Asokan et al. [20] have put forth an authentication schema predicated on a spanning-tree topology to enhance scalability and flexibility. However, its resilience to memory-based attacks has been questioned by subsequent research, such as that conducted by Ibrahim et al. [21]. Chen et al. [22] have proposed an advanced direct anonymous attestation mechanism for network-connected UAV (NC-UAV) systems that leverages trusted platform modules (TPMs) or specialized cryptoprocessors for secure credential storage. Despite its robustness, the financial overhead associated with these hardware components raises questions regarding its wide-scale applicability in commercial UAV systems. Apart from those discussed, there are few recent studies, such as those by [23] and [24], have provided enhanced security provisioning in the UAV domain and have gained widespread adoption. Nevertheless, there exists a potential for performance improvement, which we have demonstrated in the results section.

Apart from them other adopted schemes are, including those by [25] and [26], that have concentrated on scalability aspects. These studies have primarily focused on

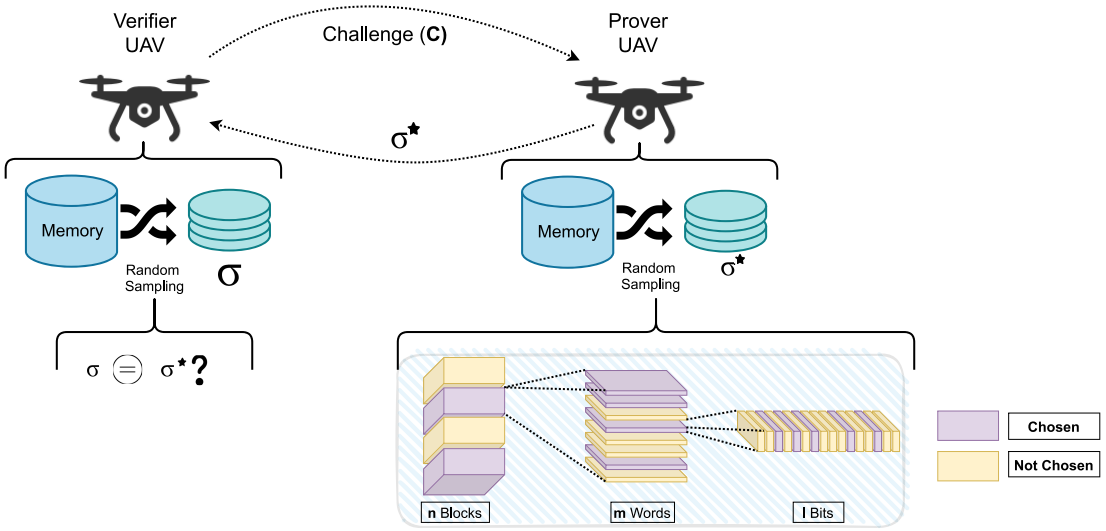


Fig. 1. Working of the peer-peer mutual authentication–attestation protocol.

authentication and attestation for multiple UAVs. However, they typically exhibit suboptimal performance in one-to-one authentication scenarios. Conversely, the schemes adopted by [25] and [26] have been directed toward scalability. These studies are predominantly geared toward authentication and attestation processes involving multiple UAVs. Nonetheless, their performance tends to be less than optimal in scenarios requiring one-to-one authentication.

Consequently, there is an imperative need for a dependable, decentralized, and scalable mechanism that combines authentication and attestation for UAV swarms. Such a mechanism should possess the capability to swiftly and efficiently authenticate all UAVs within a swarm while also ensuring safety in the physical environment.

## III. System Model

The architecture under study comprises a swarm of UAVs programmed for collective collaboration to accomplish a specific objective. A central aspect of ensuring the secure operation of this swarm lies in the ongoing verification and certification of each individual UAV. To this end, each UAV is equipped with a PUF chips. These PUF devices serve a dual role: they act as unique identifiers for each UAV and offer a measure of protection against unauthorized hardware modifications. Originating from the inherent randomness in the manufacturing process, PUFs are predicated on a challenge-response paradigm. For any given challenge ($C$), the PUF generates a corresponding unique response ($R$). Any attempt to alter or remove the PUF renders the device nonoperational, thereby precluding the compromised UAV from interacting with the BS [23], [24]. The UAVs also contain onboard memory that houses essential software components. To optimize the verification process, the verifier UAV does not inspect the entirety of the stored memory but selectively scrutinizes a representative subset during the attestation phase. Fig. 1 provides a schematic representation of this architecture, offering insights into the workflow of the authentication and attestation process.

### A. Threat and Attack Model

The security model incorporates the Dolev–Yao adversary model [15], which assumes that the attacker possesses the capability to intercept, modify, and fabricate messages within the network. This includes, but is not limited to, the launching of sophisticated network-based attacks, such as man-in-the-middle (MITM), impersonation, and message replay attacks. These attacks are designed to compromise the integrity of the communication by intercepting transmissions, mimicking authorized entities, and replaying previously intercepted messages. In addition to these network-based vulnerabilities, the model also accounts for the risk associated with the physical capture of a UAV by an adversary. In such a scenario, the attacker might tamper with the hardware to extract stored secret credentials. Given the utilization of PUFs, this kind of attack would render the UAV incapable of interacting with the BS, thereby adding an additional layer of security against hardware-based vulnerabilities.

By articulating the system and threat models in accordance with the Dolev–Yao model, this study aims to construct a comprehensive framework that addresses both network-based and hardware-based vulnerabilities.

## IV. Proposed Protocol

In this section, we describe the working of the proposed authentication–attestation protocol. Before the UAV swarm is deployed for carrying out its task, all the UAVs are deployed with a PUF and a memory. The challenge-response pairs generated by each device using its PUF are stored in the trusted server in registration process as below: 1) each UAV is registered with the BS before deployment; 2) during the registration process, a challenge-response pair from UAV's PUF is securely stored in the BS database; and 3) a device identity $ID$ is generated for each UAV at BS.

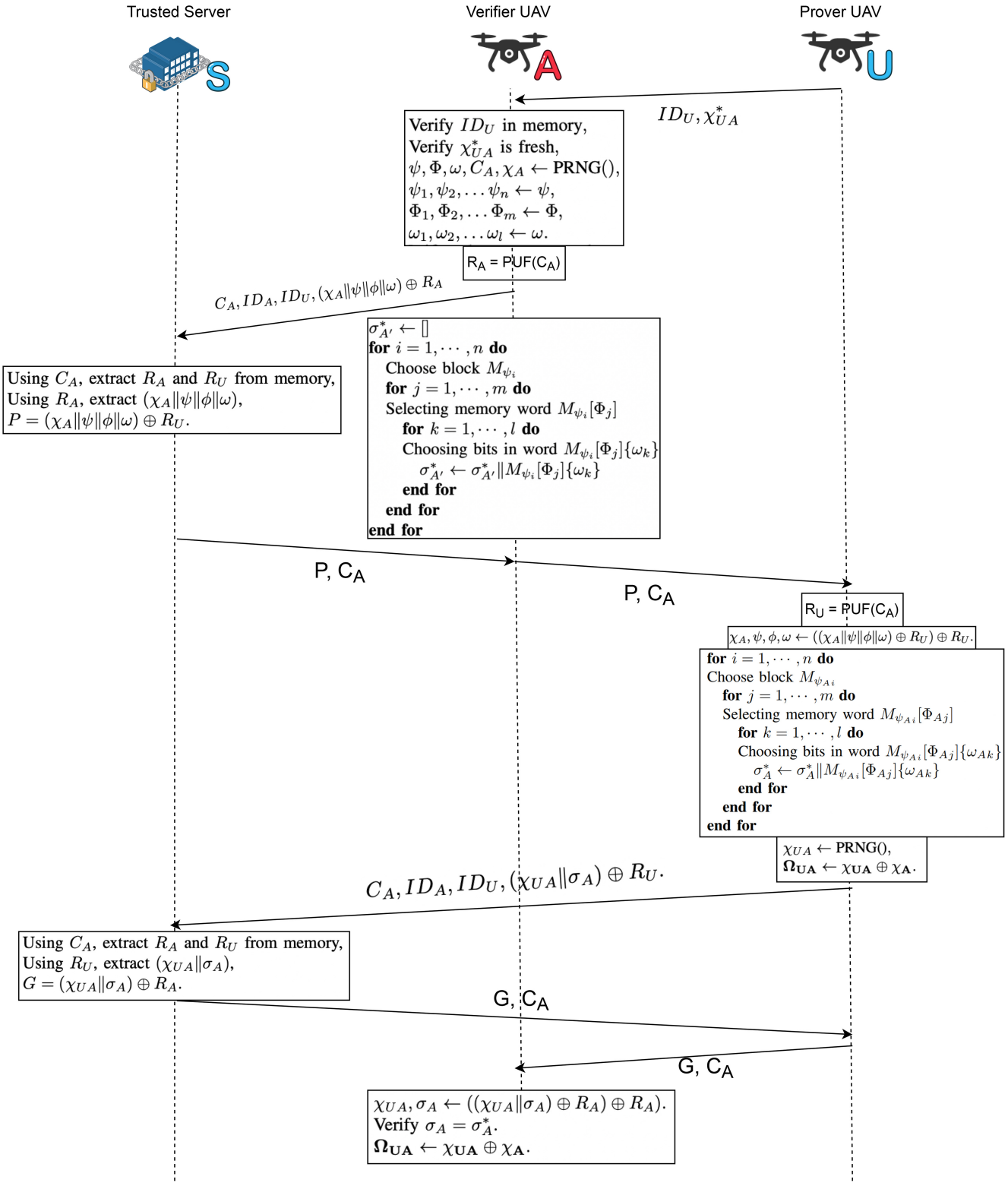


Fig. 2. Proposed authentication–attestation protocol.

Given the challenge and id of the device, the trusted server can evaluate the corresponding response. The trusted server is assumed to be secure and cannot be compromised. In all the UAVs, the same code is stored in the memory. Thus, the contents in each location in the memory (corresponding to the stored firmware) can serve as identification for all the UAVs to indicate that they belong to the same UAV swarm. Note that here we are concerned about that part of memory that stores the code. The UAV that needs to be attested is called the prover UAV, whereas the device that verifies the UAV is called a verifier. The authentication–attestation process, as shown in Fig. 2, continues as follows.

1) During the authentication process, the prover UAV, denoted as $U$, transmits its identification $ID_U$ along with a nonce $\chi^*_{UA}$ to the verifying UAV, referred to as $A$. The purpose of this transmission is for UAV $A$ to verify whether the software installed on UAV $U$ is unaltered and correct. The nonce serves the purpose of preventing replay attacks and ensuring the uniqueness of each iteration of the protocol. Upon receiving the nonce from UAV $U$, UAV $A$ checks whether the nonce is fresh or not. If the nonce is determined to be stale or previously used, the message containing the nonce is discarded and the authentication process is halted to prevent any potential security breaches. To avoid storing all the nonce, the receiver maintains a window or cache of recently used nonces. When a message with a nonce is received, the receiver checks if the nonce is within an acceptable range or if it falls within the window of recently used nonces. If the nonce is within the acceptable range, it is considered fresh; otherwise, it is rejected as a potential replay.
2) Additionally, the verifying UAV $A$ generates its own nonce $\chi_A$ using a pseudo-random generator (PRNG). This generated nonce $\chi_A$ plays a crucial role in the authentication process, as it is utilized to derive the session key after the authentication step is completed. The session key is an important cryptographic element that enables secure communication between the prover UAV ($U$) and the verifying UAV ($A$) following successful authentication.
3) The verifier UAV utilizes a PRNG to generate a challenge $C_A$, a nonce $\chi_A$, and three random sets: 1) $\psi$; 2) $\phi$; and 3) $\omega$

$$\begin{aligned} C_A &= \text{PRNG}(), \\ \chi_A &= \text{PRNG}(), \\ \psi = \text{PRNG}(), \quad \phi &= \text{PRNG}(), \quad \omega = \text{PRNG}(). \end{aligned}$$

4) After generating the required random sets, the verifier then computes the attestation value $\sigma^*_{A'}$ using randomly selected memory locations. In particular, UAV $A$ iterates through all the bits $[\omega_1, \omega_l]$ within the $m$ words $[\Phi_1, \Phi_m]$ of the selected blocks $[\psi_1, \psi_n]$. These bits are concatenated to form the value $\sigma^*_{A'}$, which can be calculated as

$$\sigma^*_{A'} = \left( \bigoplus_{i=1}^{n} \bigoplus_{j=1}^{m} \bigoplus_{k=1}^{l} \omega_k \right)$$

where $\bigoplus$ represents the XOR operation. The pseudo algorithm is presented in Algorithm 1.
5) Using its PUF, UAV $A$ evaluates the response $R_A$ corresponding to the challenge $C_A$ that it generated. The response $R_A$ is XORed with $(\chi_A \| \psi \| \phi \| \omega)$ and sent to the trusted server $S$, along with the challenge $C_A$, ID of verifier UAV $ID_A$, and ID of prover UAV $ID_U$

$$R_A \oplus (\chi_A \| \psi \| \phi \| \omega).$$

**Algorithm 1:** Evaluation of $\sigma^*_{A'}$

**Result**: $\sigma^*_{A'}$
**Initialisation**:$\sigma^*_{A'} \leftarrow []$
Loop over i where $i$ = 1 to $n$
  **Select** $M_{\psi_i}$
  Loop over j where $j$ = 1 to $m$
  **Select** $M_{\psi_i}[\Phi_j]$
    Loop over k where $k$ = 1 to $l$
    **Select** $M_{\psi_i}[\Phi_j]\{\omega_k\}$
      $\sigma^*_{A'} \leftarrow \sigma^*_{A'} \| M_{\psi_i}[\Phi_j]\{\omega_k\}$

6) The trusted server stores the CRP responses for each device. Thus, the trusted server extracts the $R_A$ corresponding to $C_A$ for UAV $A$ from its memory. Using $R_A$, it extracts $(\chi_A \| \psi \| \phi \| \omega)$ by performing an XOR operation on $(\chi_A \| \psi \| \phi \| \omega) \oplus R_A$ with $R_A$.
7) Then, the trusted server computes $R_U$ corresponding to $C_A$ for device $U$ from its memory. Using $R_U$, it generates the reply message $P$ by applying the XOR operation between $(\chi_A \| \psi \| \phi \| \omega)$ and $R_U$. Finally, it sends the reply message $P$ to UAV $A$ along with $C_A$

$$P = (\chi_A \| \psi \| \phi \| \omega) \oplus R_U.$$

8) The verifier $A$, on receiving reply $P$ from the trusted server, forwards the message to the prover UAV $U$.
9) On receiving the message ($P$, $C_A$) from device $A$, UAV $U$ uses $C_A$ to generate $R_A$ using its PUF.
10) After UAV $U$ generates its attestation value ($\sigma^*_A$), it proceeds to generate an additional nonce $\chi UA$ through the utilization of PRNG. With $\chi_{UA}$ and $\chi_A$, UAV $U$ generates the session key $\Omega_{UA}$ according to the following expression:

$$\Omega_{\mathbf{UA}} \leftarrow \chi_{\mathbf{UA}} \oplus \chi_{\mathbf{A}}.$$

11) The UAV $U$ creates a message $(\chi_{UA} \| \sigma_A)$ and applies an XOR operation with $R_U$ before sending the resulting message $(\chi_{UA} \| \sigma_A) \oplus R_U$ to the trusted server. Along with this message, UAV $U$ also includes the challenge $C_A$, as well as the IDs of the verifier UAV ($ID_A$) and the prover UAV ($ID_U$).
12) From the memory, the trusted server retrieves $R_U$ associated with $C_A$ for UAV $U$. Utilizing $R_U$, it extracts $(\chi_{UA} \| \sigma_A)$ by performing an XOR operation on $(\chi_{UA} \| \sigma_A) \oplus R_U$ with $R_U$. Subsequently, the trusted server retrieves $R_A$ corresponding to $C_A$ for device $A$ from its memory. Using $R_A$, it generates the reply message $G$ by applying an XOR operation between $(\chi_{UA} \| \sigma_A)$ and $R_A$. Finally, the trusted server sends the reply message $G$ to UAV $U$ along with $C_A$.
13) Upon receiving the reply $G$ from the trusted server, the prover UAV $U$ proceeds to transmit the message to the verifier UAV $A$. Subsequently, upon receiving the message $G$ from $U$, the verifier UAV $A$ utilizes the PUF response $R_A$ to extract the values $\chi_{UA}$ and $\sigma_A$. It then proceeds to compare the attestation value $\sigma^*_A$ with its computed attestation value $\sigma_A$. If the two values match,

it indicates successful attestation of the prover, and the session key $\Omega_{UA}$ is generated as follows:

$$\Omega_{UA} \leftarrow \chi_{UA} \oplus \chi_A.$$

## V. Security Analysis

We prove the secrecy of the proposed protocol by verifying that the secrets, such as $\Omega_{UA}$, $\chi_A$, and $\chi_{UA}$, are exclusively known only to UAV $A$ and UAV $U$. Recall that the trusted server is assumed to be secure and cannot be compromised. The knowledge of secrets by a trusted server $S$ is not considered a vulnerability. Thus, when we claim that no entity other than $A$ knows the secret, the idea is that no UAV, other than $A$ and the trusted server, knows the secret. The proof is presented using the inference rules of Mao and Boyd's logic [27]. The rules used in this proof are presented in Table I. Here, the notations of variables and symbols are the same as used by Mao and Boyd [27].

We begin with "UAV $U$ is convinced that $\chi_{UA}$ is a secret shared among only $U$, $S$, and $A$. No other entity knows this secret." In case UAV $A$ is authentic, the PUF response $R_A$ generated by UAV $A$ is the same as $R_A$ stored in the memory of the trusted server $S$. Similarly, if UAV $U$ is authentic, then the PUF response $R_U$ generated by UAV $U$ is the same as $R_U$ stored in the memory of the trusted server $S$. Thus, we can formulate

$$U \mid\equiv U \overset{R_U}{\leftrightarrow} S, \tag{i(a)}$$

$$S \mid\equiv S \overset{R_A}{\leftrightarrow} A. \tag{i(b)}$$

Using (i(a)) and (i(b))

$$U \mid\equiv U \overset{R_A S_{R_U}}{\leftrightarrow} A. \tag{i}$$

UAV $U$ decrypts the message from $S$ using $R_U$, where $S$ had decrypted the message from $A$ using $R_A$. We denote this decryption key as ${}_{R_A}S_{R_U}$. $S$ decrypts using $R_A$ and encrypts using $R_U$. From the protocol, it can be observed that UAV $U$ can identify the value of $\chi_A$ by performing a XOR operation between $P$ and $R_U$, and S can obtain $\chi_A$ using $R_A$ stored in the memory

$$U \overset{R_U}{\triangleleft} \chi_A, \tag{ii(a)}$$

$$S \overset{R_A}{\triangleleft} \chi_A. \tag{ii(b)}$$

By considering the security and unclonability of a PUF, it can be concluded that only UAV $U$ and server $S$ have the capability to generate the correct shared value $R_U$. Consequently, we can state that "$U$ believes that $S$ has encrypted $\chi_A$ using the key $R_U$." This inference is derived by applying the *authentication rule* to the statements (i(a)) and (ii(a))

$$U \mid\equiv S \overset{R_U}{|\sim} \chi_A. \tag{iii(a)}$$

Similarly, applying the *authentication rule* to statements (i(b)) and (ii(b)) we get

$$S \mid\equiv A \overset{R_A}{|\sim} \chi_A. \tag{iii(b)}$$

TABLE I
Mao Boyd Authentication Rules

| Name | Inference Rule |
|---|---|
| **Authentication rule** | $\frac{\mathcal{A} \mid\equiv \mathcal{A} \overset{K}{\leftrightarrow} \mathcal{B} \wedge \mathcal{A} \overset{K}{\triangleleft} M}{\mathcal{A} \mid\equiv \mathcal{B} \overset{K}{\mid\sim} M}$ |
| **Nonce-verification rule** | $\frac{\mathcal{A} \mid\equiv \#(M) \wedge \mathcal{A} \mid\equiv \mathcal{B} \overset{K}{\mid\sim} M}{\mathcal{A} \mid\equiv \mathcal{B} \mid\equiv \mathcal{A} \overset{K}{\leftrightarrow} \mathcal{B}}$ |
| **Confidentiality rule** | $\frac{\mathcal{A} \mid\equiv \mathcal{A} \overset{K}{\leftrightarrow} \mathcal{B} \wedge \mathcal{A} \mid\equiv S^c \triangleleft \| M \wedge \mathcal{A} \overset{K}{\mid\sim} M}{\mathcal{A} \mid\equiv (S \cup \{\mathcal{B}\})^c \triangleleft \| M}$ |
| **Super-principal rule** | $\frac{\mathcal{A} \mid\equiv \mathcal{B} \mid\equiv X \wedge \mathcal{A} \mid\equiv sup(\mathcal{B})}{\mathcal{A} \mid\equiv X}$ |
| **Intuitive rule** | $\frac{\mathcal{A} \overset{K}{\triangleleft} M}{\mathcal{A} \triangleleft M}$ |
| **Good Key rule** | $\frac{\mathcal{A} \mid\equiv \{\mathcal{A}, \mathcal{B}\}^c \triangleleft \| K \wedge \mathcal{A} \mid\equiv \#(K)}{\mathcal{A} \mid\equiv \mathcal{A} \overset{K}{\leftrightarrow} \mathcal{B}}$ |
| **Fresh rule** | $\frac{\mathcal{A} \mid\equiv \#(M) \wedge \mathcal{A} \triangleleft N \mathbf{R} M}{\mathcal{A} \mid\equiv \#(N)}$ |

Thus, using (iii(a)) and (iii(b))

$$U \mid\equiv A \overset{R_A S_{R_U}}{|\sim} \chi_A. \tag{iii}$$

Due to the fact that each iteration of the protocol is unique, and UAV $A$ produces a new nonce $\chi_A$ each time, we can conclude that $S$ knows $\chi_A$ is fresh. Thus, $U$ also knows $\chi_A$ is fresh

$$U \mid\equiv \#(\chi_A). \tag{iv}$$

By using the *nonce-verification rule* on (iii) and (iv), we may verify that $U$ is convinced that $A$ is certain that $\chi_A$ is a well-kept secret between $A$, $S$ and itself as

$$U \mid\equiv S \mid\equiv U \overset{\chi_A}{\leftrightarrow} S, \tag{v(a)}$$

$$S \mid\equiv A \mid\equiv S \overset{\chi_A}{\leftrightarrow} A. \tag{v(b)}$$

From (v(a)) and (v(b))

$$U \models A \models U \overset{\chi_A}{\leftrightarrow} A. \tag{v}$$

Since UAV $U$ generates the nonce $\chi_{UA}$, we can write $U$ sees $\chi_{UA}$ without any decipherment key as

$$U \triangleleft \chi_{UA}. \tag{vi}$$

Applying the *Intutive rule*, we can obtain $U$ sees $\chi_A$ with decipherment key $R_U$

$$U \overset{R_A S_{R_U}}{\triangleleft} \chi_{UA}. \tag{vii}$$

By applying the *authentication rule* on (i) and (vii), we get $U$ is convinced that $A$ encrypted $\chi_A$ using $R_A$ and then S encrypted using $R_U$

$$U \models A \overset{R_A S_{R_U}}{|\sim} \chi_A. \tag{viii}$$

Because $A$ produces a fresh nonce $\chi_A$ each time, $U$ is certain that no one other than $S$ and $A$ has seen $\chi_A$

$$U \models A \models \{U, S\}^c \triangleleft \| \chi_A. \tag{ix}$$

By applying the *confidentiality rule* to (v), (viii), and (ix), we get (x), which says that $U$ is convinced that $A$ is sure that no one but $U$, $S$, and $A$ has access to $\chi_A$

$$U \models A \models \{U, S, A\}^c \triangleleft \| \chi_A. \tag{x}$$

Since $A$ is the verifier and $U$ is the prover, $U$ believes that $A$ is a super-principal or credible verifier

$$U \models \sup(S). \tag{xi}$$

Next, using *super-principal rule* on (x) and (xi), we obtain

$$U \models \{U, A\}^c \triangleleft \| \chi_A. \tag{xii}$$

$U$ transmits $\chi^*_{UA}$ to $A$ in the first message of Fig. 2. $A$ responds in the second message by sending $\chi_A$. $A$ gets the nonces $\chi^*_{UA}$ and $\chi_A$ via sending to S. S deciphers with $R_A$, while encrypts with $R_U$, which $U$ can decrypt using its PUF. As a result, $\chi^*_{UA}$ may be regarded as a challenge, whereas $\chi_A$ can be regarded as a response, according to the message idealization criteria. Thus, we arrive at the statement that $U$ may view the responded challenge $\chi^*_{UA}$ and the response $\chi_A$ with decryption key $_{R_A}S_{R_U}$

$$U \overset{R_A S_{R_U}}{\triangleleft} \chi^*_{UA} \ \mathbf{R} \ \chi_A. \tag{xiii}$$

By applying the *intuitive rule* to (xiii), we get $U$ may see the responded challenge $\chi^*_{UA}$ and response $\chi_A$

$$U \triangleleft \chi^*_{UA} \ \mathbf{R} \ \chi_A. \tag{xiv}$$

Next, we prove the statement, "U believes that $A$'s shared key, $\chi_A$, is valid." We use the *good-key rule* on (iv), (xi), and (xii), to get

$$U \models U \overset{\chi_A}{\leftrightarrow} A. \tag{xv}$$

Similarly, we can prove that UAV $A$ and UAV $U$ are convinced that all other secrets, such as $\Omega_{UA}$, $\chi_A$, and $\chi_{UA}$ are exclusively known only to UAV $A$ and UAV $U$, as shown in Fig. 3.

For an attack scenario, consider an adversary, denoted as $E$, attempts to impersonate the trusted server $S$ to send malicious commands to UAV $U$. The adversary tries to generate a message that mimics being from $S$ by using a compromised key or fabricating a message with guessed authentication parameters. Both UAV $U$ and server $S$ share unique PUF-derived keys $R_U$ and $R_A$, which are securely stored and unknown to the adversary $E$. This proof demonstrates that our UAV communication protocol effectively thwarts advanced spoofing attacks through a series of cryptographic checks and balances, ensuring that all communications are authentic, fresh, and secure.

## VI. Results and Discussion

In our comprehensive performance evaluation, we focused on a detailed analysis of our proposed protocol, utilizing the Raspberry Pi 3B board as a surrogate for UAV computational systems. This board is equipped with a 1.2 GHz 64-bit quad-core ARM Cortex-A53 processor, supported by 1GB LPDDR2 RAM, and we used a 32GB Class 10 MicroSD card for storage. The entire setup ran on Raspbian OS, with experiments conducted in Python 3.7 due to its balance of performance and ease of use. We meticulously recorded the computational performance across various operations, such as encompassing XOR, PRNG, SHA-1 hash functions, HMAC leveraging SHA-1, and sequential concatenations. For instance, HMAC generation using SHA-1, conducted over 256-bit keys. The PUF technology integrated, as detailed in [28], boasted an impressive 320-bit output with an operational speed of just 0.4 $\mu$s, affirming its suitability for high-security applications in constrained environments.

Fig. 4 offers a side-by-side comparison of the total time consumption for our protocol against those in [12], [13], [15], [23], and [24]. A nuanced analysis reveals that while competing methods have computational costs fluctuating between 355 $\mu$s and a peak of 585 $\mu$s, our protocol stands out, clocking in at a mere 182 $\mu$s. It is essential to note that none of the competing protocols can simultaneously offer both authentication and attestation.

In our study, a comprehensive analysis of the timings associated with various cryptographic operations was conducted, and these were compared against data from multiple sources. Our findings, as summarized in Table II, provide a detailed overview of the time efficiencies across different cryptographic procedures.

We observed significant variations in the execution times for operations, such as bitwise XOR (64-bit), addition (64-bit), and multiplication (64-bit binary). For instance, the Bitwise XOR operation showcased a timing range from as low as 0.00 $\mu$s in one study to a high of 252 $\mu$s in another. Similarly, Encryption and Decryption operations using AES (32-Bytes) also revealed interesting disparities. While some sources reported no time consumption 0.00 $\mu$s, others observed times up to 246 $\mu$s. This indicates a considerable disparity in performance, potentially due to differences in computational resources or implementation techniques. Particularly noteworthy is the Hash SHA256 (64-Bytes) operation and AES Encryption operations, where

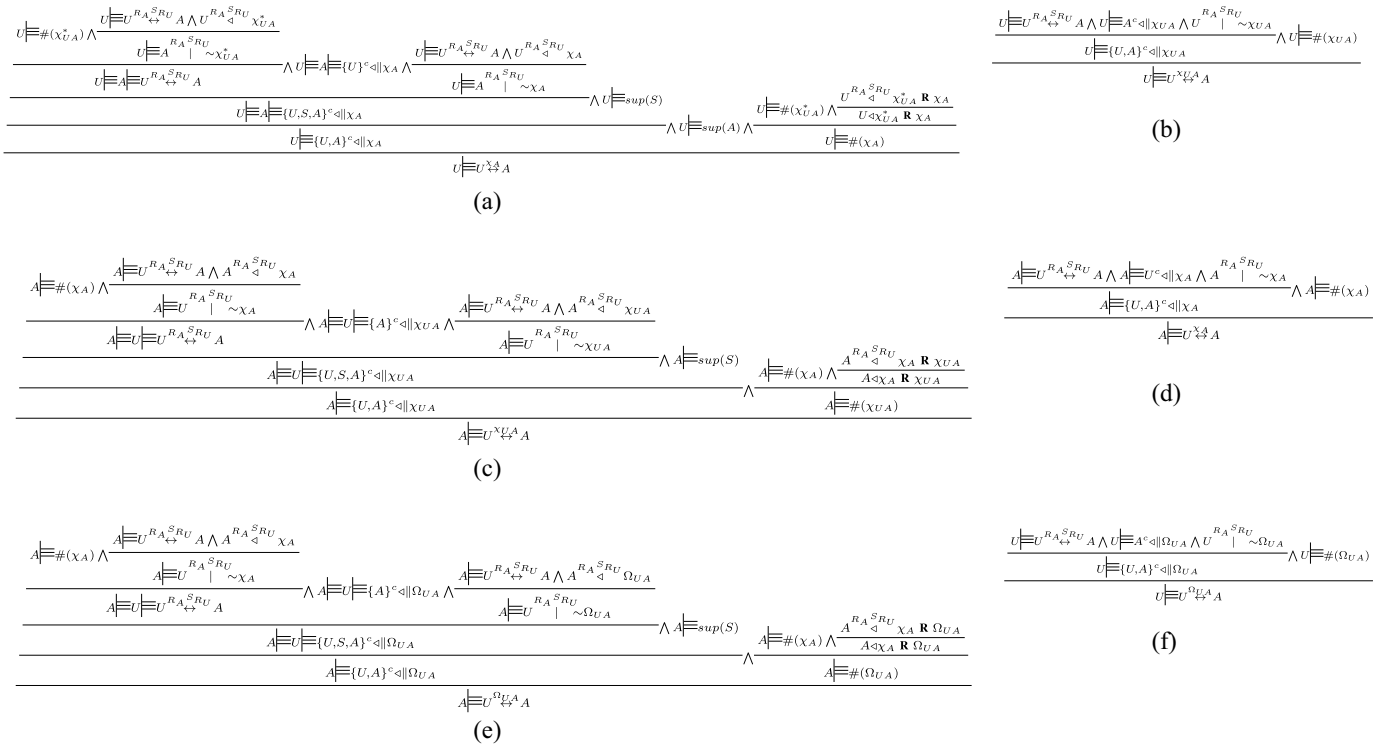

Fig. 3. Mao–Boyd proofs for the proposed protocol between UAV U and UAV *A*. (a) $\chi_A$ is a secure secret between "*U*" and "*A*," according to the conviction of *U*. (b) $\chi_{UA}$ is a secure secret between *U* and *A*, according to the conviction of *U*. (c) $\chi_{UA}$ is a secure secret between *U* and *A*, according to the conviction of *A*. (d) $\chi_A$ is a secure secret between *U* and *A*, according to the conviction of *A*. (e) $\Omega_{UA}$ is a secure secret between *U* and *A*, according to the conviction of *A*. (f) $\Omega_{UA}$ is a secure secret between *U* and *A*, according to the conviction of *U*.

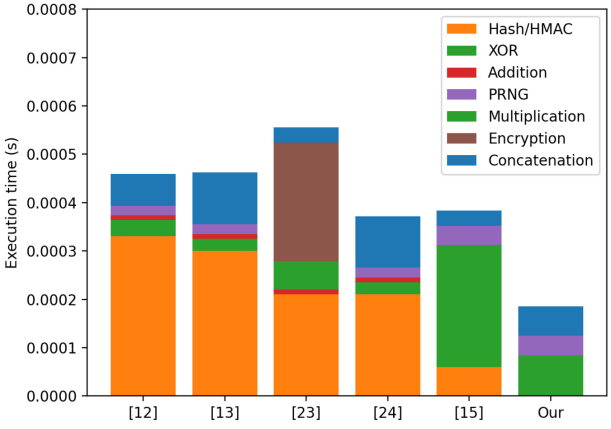


Fig. 4. Comparative analysis of total execution time.

a significant reduction in time was noted. Our study observed times notably shorter than those reported in other studies, such as 390 and 66 $\mu$s. The implementation of PUFs showed minimal time consumption across most studies, with our results aligning with the fastest reported time of 0.8 $\mu$s, illustrating the efficiency of PUFs in cryptographic operations. Cumulatively, the total computation time for all operations in our study was calculated to be 182 $\mu$s, markedly lower than the timings observed in other referenced studies, which ranged from 355 to 585 $\mu$s. This substantial reduction in our total computation time not only signifies the effectiveness of the cryptographic techniques employed in our study but also highlights the potential for further optimizations in the field. We have included performance of [25] and [26], to showcase their computational expensive nature in one-one authentication.

Fig. 5 presents a detailed comparative analysis of various protocols, focusing on their computational overhead and energy efficiency. The protocol referenced in [12] utilized approximately 3.05 $\times$ $10^6$ CPU cycles for its execution, resulting in an energy consumption of about 7.77 mJ. In contrast, the protocol introduced by [13] required a higher computational effort of nearly 3.4 $\times$ $10^6$ CPU cycles, leading to an energy expenditure of nearly 8.4 mJ. The protocol from [23] demanded even more substantial resources, using around 5.42 $\times$ $10^6$ CPU cycles and consuming 12.82 mJ of energy, marking it as one of the more energy-intensive options compared here. Additionally, the approach by [24], showed an energy requirement of 5.8 mJ and computational needs of 2.5 $\times$ $10^6$ cycles, while [15] consumed 7.8 mJ and required 3 $\times$ $10^6$ cycles. Protocols [25] and [26], used substantial resources requiring more than 10 $\times$ $10^6$ cycles and 24 mJ of energy. Our proposed protocol demonstrated significant enhancements in efficiency. It needed only 1.67 $\times$ $10^6$ CPU cycles, illustrating its focus on computational efficiency and making it an excellent candidate for UAV-based applications with its energy consumption at a mere 3.8 mJ.

In terms of scalability, earlier papers were not really designed for handling multiple UAVs efficiently. The choice to highlight [15] over [25] and [26] in Fig. 4 was based on [15] claiming better computational performance in single

TABLE II
TIMINGS OF DIFFERENT CRYPTOGRAPHIC OPERATIONS

| Cryptographic Operations | [12] | [13] | [23] | [24] | [15] | [25] | [26] | Our |
|---|---|---|---|---|---|---|---|---|
| Bitwise XOR (64-bit) | 3.37E-05 | 2.52E-05 | 0.00E+00 | 2.52E-05 | 2.52E-04 | 2.52E-05 | 2.52E-05 | 1.68E-05 |
| Addition(64-bit) | 9.64E-06 | 9.64E-06 | 9.64E-06 | 9.64E-06 | 0.00E+00 | 9.64E-06 | 0.00E+00 | 0.00E+00 |
| Multiplication (64-bit binary) | 0.00E+00 | 0.00E+00 | 5.90E-05 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| PRNG | 2.03E-05 | 2.03E-05 | 0.00E+00 | 2.03E-05 | 4.06E-05 | 8.06E-05 | 4.06E-05 | 4.06E-05 |
| Hash SHA256 (64-Bytes) | 3.90E-04 | 4.20E-04 | 2.70E-04 | 3.00E-04 | 6.60E-05 | 8.40E-04 | 6.00E-04 | 0.00E+00 |
| Encryption/Decryption(AES) (32-Bytes) | 0.00E+00 | 0.00E+00 | 2.46E-04 | 0.00E+00 | 2.46E-04 | 4.92E-04 | 4.92E-04 | 1.23E-04 |
| PUF | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 8.00E-07 | 0.00E+00 | 1.60E-06 | 8.00E-07 |
| **Total computation time (s)** | **4.54E-04** | **4.75E-04** | **5.85E-04** | **3.55E-04** | **4.14E-04** | **14.37E-04** | **11.60E-04** | **1.82E-04** |

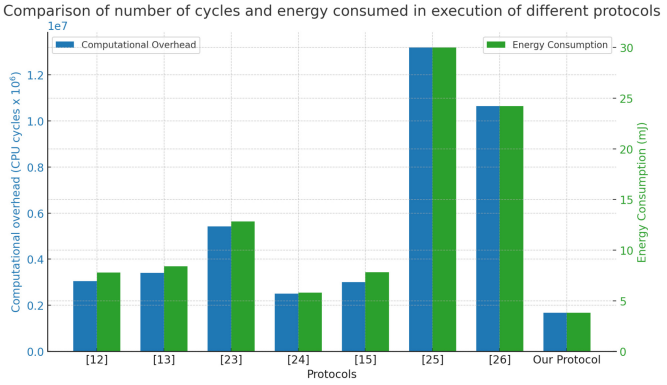


Fig. 5. Comparison of number of cycles and energy consumed in execution of different protocols (mJ).

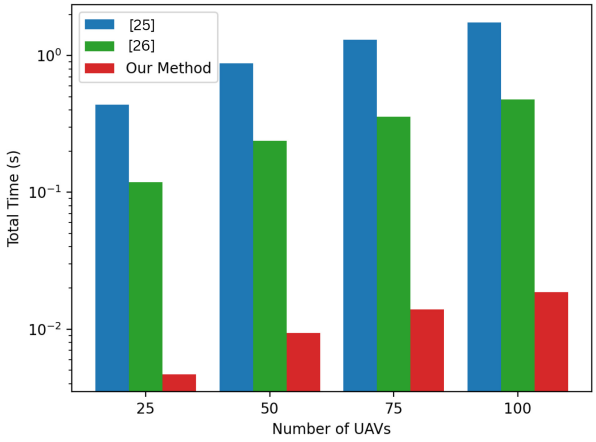


Fig. 6. Comparison of total execution time with respect to the number of UAVs.

UAV authentication. However, as we scale up to authenticate multiple UAVs (25–100), Alladi et al. [15] started to need more computational resources. In this context, Kohnhäuser et al. [25] and Alladi et al. [26] showed superior performance. This adjustment in the comparison takes into account the different needs and efficiencies when dealing with authentication for a group of UAVs. It is a recognition that the best choice depends on the scale and context of the authentication scenario.

In our study, to effectively compare the scalability and efficiency of our protocol, we chose to benchmark against protocols [25], [26]. This decision is based on the fact that the previously compared works, Wazid et al. [12], Srinivas et al. [13], Alladi et al. [15], Ali et al. [23], and Verma et al. [24] primarily focused on single authentication mechanisms. Unlike these works, Kohnhäuser [25] and Alladi [26] provided a broader framework by incorporating both scalability and attestation features, which are crucial for the comprehensive evaluation of our protocol in the context of UAV swarm operations.

Further, Fig. 6 illustrates the impact of varying the number of UAVs on the total execution time, with a specific comparison against [25], [26]. These two works stand out as they offer both authentication and attestation features. Our investigation reveals that while [25] needed 0.219 s for operating 25 UAVs and 0.875 s for 100 UAVs, our protocol astonishingly required just 4650 $\mu$s in both instances. For [26], the execution time was 595 $\mu$s for 25 UAVs and 0.0238 s for 100 UAVs. This proves that our protocol's efficiency improves relative to [25] and [26] as the number of UAVs grows.

Finally, Fig. 7 examines how varying memory sizes for attestation influence the total attestation time of our protocol, contrasting it with [25] and [26]. As the number of memory locations requiring validation increases, so does the computational time. The data indicates that the time taken to attest 128 KB of memory using our protocol stands at 0.374 ms, while it shoots up to approximately 3 ms for 1 MB. On the other hand, the protocol in [26] needed 4.76 ms to attest 128 KB and 53.02 ms for 1024 KB. The least efficient was [25], which clocked 17.5 ms for 128 KB and 144.1 ms for 1024 KB. These findings affirm that our protocol not only scales effectively but also becomes more efficient relative to [25] and [26] as the memory size for attestation increases.

## VII. PRACTICAL CHALLENGES IN DEPLOYMENT OF OUR PROTOCOL

One of the key challenges in deploying our peer-to-peer security protocol for UAV swarms lies in ensuring hardware and software compatibility. The protocol must seamlessly integrate with a variety of UAV hardware and software configurations, a task made complex by the continuous evolution and diversity of UAV technology. Furthermore, the protocol must robustly handle the variability in network connectivity inherent in UAV swarms, where factors, such as environmental conditions, UAV movements, and interference, can greatly affect communication. Integrating this protocol with existing UAV control and communication systems poses another significant challenge, especially when these systems

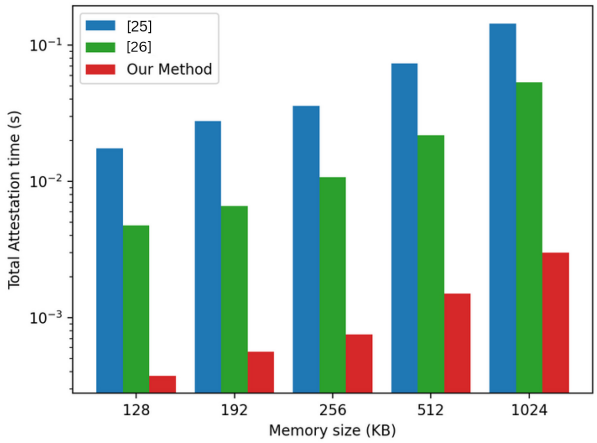


Fig. 7. Comparison of overall attestation time against memory size (KB).

have their own proprietary security measures or protocols. Finally, compliance with a range of regulatory standards and privacy laws, which can vary across different countries and regions, is crucial. This aspect adds an additional layer of complexity to the protocol's deployment, necessitating careful consideration to ensure adherence to all relevant legal and regulatory requirements.

## VIII. Current Assumptions and Future Work

This section revisits the assumptions made in the current design of UAV communication protocols and discusses potential areas for future improvement, ensuring these systems meet the escalating demands of modern UAV applications.

1) *Computational Efficiency:*
   a) *Assumption:* Traditional cryptographic mechanisms assume high-computational capabilities within UAVs, which may not be feasible for smaller or less capable UAVs.
   b) *Future Work:* Research should focus on integrating lightweight cryptographic algorithms that reduce computational load, enhancing the system's overall efficiency and responsiveness.
2) *Robustness against Dynamic AI Threats:*
   a) *Assumption:* Security measures are designed for static threat models without consideration for evolving or sophisticated cyber threats arising for progress in AI.
   b) *Future Work:* Develop adaptive security mechanisms that can detect and respond to real-time threats, increasing the resilience of UAV communications against advanced cyber-attacks.
3) *Energy Consumption:*
   a) *Assumption:* It is assumed that UAVs have ample energy to support continuous security operations, which may not hold true in extended missions.
   b) *Future Work:* Prioritize the creation of energy-efficient protocols that conserve battery life while maintaining high-security standards, thus extending the operational duration of UAVs.

Addressing these assumptions and focusing on the outlined future work will significantly enhance the practicality and security of UAV communication protocols, meeting both current and future operational demands.

## IX. Conclusion

UAVs have garnered significant attention for their multi-faceted applications in social, economic, and military domains. However, their operational efficacy is heavily contingent upon secure wireless communication infrastructures. In this work, we have introduced a lightweight yet robust authentication and attestation framework tailored for UAV swarms. Through rigorous analysis based in Mao and Boyd's logical framework, we have established the feasibility, scalability, and security efficacy of our proposed protocol. Empirical results further substantiate that our approach not only meets but also exceeds the performance metrics of existing protocols, particularly in terms of execution time. Our work, therefore, serves as a promising avenue for ensuring secure and efficient communications in large-scale UAV deployments.

## References

[1] T. Alladi, Naren, G. Bansal, V. Chamola, and M. Guizani, "SecAuthUAV: A novel authentication scheme for UAV-ground station and UAV-UAV communication," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 15068–15077, Dec. 2020.
[2] H. Menouar, I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer, "UAV-enabled intelligent transportation systems for the smart city: Applications and challenges," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 22–28, Mar. 2017.
[3] M. McNabb. "Counter drone technology: D-fend solutions on the major incidents and developments of 2022, and what to expect this year." 2023. Accessed: May 18, 2023. [Online]. Available: https://dronelife.com/
[4] G. Bansal, N. Naren, V. Chamola, B. Sikdar, N. Kumar, and M. Guizani, "Lightweight mutual authentication protocol for V2G using physical unclonable function," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7234–7246, Jul. 2020.
[5] M. Burrows, M. Abadi, and R. M. Needham, "A logic of authentication," *Proc. Roy. Soc. London. A. Math. Phys. Sci.*, vol. 426, no. 1871, pp. 233–271, 1989.
[6] E. Brickell, J. Camenisch, and L. Chen, "Direct anonymous attestation," in *Proc. 11th ACM Conf. Comput. Commun. Secur.*, 2004, pp. 132–145.
[7] G. Coker et al., "Principles of remote attestation," *Int. J. Inf. Security*, vol. 10, no. 2, pp. 63–81, 2011.
[8] Y. Zeng, S. Jin, Q. Wu, and F. Gao, "Network-connected UAV communications," *China Commun.*, vol. 15, no. 5, pp. 111–121, Dec. 2018.
[9] A. Fotouhi et al., "Survey on UAV cellular communications: Practical aspects, standardization advancements, regulation, and security challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3417–3442, 4th Quart., 2019.
[10] X. Sun, D. W. K. Ng, Z. Ding, Y. Xu, and Z. Zhong, "Physical layer security in UAV systems: Challenges and opportunities," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 40–47, Oct. 2019.
[11] A. Birk, B. Wiggerich, H. Bülow, M. Pfingsthorn, and S. Schwertfeger, "Safety, security, and rescue missions with an unmanned aerial vehicle (UAV)," *J. Intell. Robot. Syst.*, vol. 64, no. 1, pp. 57–76, 2011.
[12] M. Wazid, A. K. Das, N. Kumar, A. V. Vasilakos, and J. J. Rodrigues, "Design and analysis of secure lightweight remote user authentication and key agreement scheme in Internet of Drones deployment," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3572–3584, Apr. 2019.
[13] J. Srinivas, A. K. Das, N. Kumar, and J. J. Rodrigues, "TCALAS: Temporal credential-based anonymous lightweight authentication scheme for Internet of Drones environment," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 6903–6916, Jul. 2019.

[14] C. Jiang, Y. Fang, P. Zhao, and J. Panneerselvam, "Intelligent UAV identity authentication and safety supervision based on behavior modeling and prediction," *IEEE Trans. Ind. Informat.*, vol. 16, no. 10, pp. 6652–6662, Oct. 2020.
[15] T. Alladi, V. Chamola, Naren, N. Kumar, "PARTH: A two-stage lightweight mutual authentication protocol for UAV surveillance networks," *Comput. Commun.*, vol. 160, pp. 81–90, Jul. 2020.
[16] M. Yahuza, M. Y. I. Idris, A. W. A. Wahab, T. Nandy, I. B. Ahmedy, and R. Ramli, "An edge assisted secure lightweight authentication technique for safe communication on the Internet of Drones network," *IEEE Access*, vol. 9, pp. 31420–31440, 2021.
[17] X. Du, Y. Li, S. Zhou, and Y. Zhou, "ATS-LIA: A lightweight mutual authentication based on adaptive trust strategy in flying ad-hoc networks," *Peer-to-Peer Netw. Appl.*, vol. 15, no. 4, pp. 1979–1993, 2022.
[18] T. D. Khanh et al., "TRA: Effective authentication mechanism for swarms of unmanned aerial vehicles," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, 2020, pp. 1852–1858.
[19] E. Chen, J. Chen, A. W. Mohamed, B. Wang, Z. Wang, and Y. Chen, "Swarm intelligence application to UAV aided IoT data acquisition deployment optimization," *IEEE Access*, vol. 8, pp. 175660–175668, 2020.
[20] N. Asokan et al., "SEDA: Scalable embedded device attestation," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Security*, 2015, pp. 964–975.
[21] A. Ibrahim, A.-R. Sadeghi, G. Tsudik, and S. Zeitouni, "DARPA: Device attestation resilient to physical attacks," in *Proc. 9th ACM Conf. Security Privacy Wireless Mobile Netw.*, 2016, pp. 171–182.
[22] L. Chen, S. Qian, M. Lim, and S. Wang, "An enhanced direct anonymous attestation scheme with mutual authentication for network-connected UAV communication systems," *China Commun.*, vol. 15, no. 5, pp. 61–76, 2018.
[23] Z. Ali, S. A. Chaudhry, M. S. Ramzan, and F. Al-Turjman, "Securing smart city surveillance: A lightweight authentication mechanism for unmanned vehicles," *IEEE Access*, vol. 8, pp. 43711–43724, 2020.
[24] G. K. Verma, B. Singh, N. Kumar, and D. He, "CB-PS: An efficient short-certificate-based proxy signature scheme for UAVs," *IEEE Syst. J.*, vol. 14, no. 1, pp. 621–632, Mar. 2020.
[25] F. Kohnhäuser, D. Püllen, and S. Katzenbeisser, "Ensuring the safe and secure operation of electronic control units in road vehicles," in *Proc. IEEE Security Privacy Workshops (SPW)*, 2019, pp. 126–131.
[26] T. Alladi, S. Chakravarty, V. Chamola, and M. Guizani, "A lightweight authentication and attestation scheme for in-transit vehicles in IoV scenario," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 14188–14197, Dec. 2020.
[27] W. Mao and C. Boyd, "Towards formal analysis of security protocols," in *Proc. Comput. Secur. Found. Workshop VI*, 1993, pp. 147–158. [Online]. Available: http://ieeexplore.ieee.org/document/246631/
[28] X. Zhao, Q. Zhao, Y. Liu, and F. Zhang, "An ultracompact switching-voltage-based fully reconfigurable RRAM PUF with low native instability," *IEEE Trans. Electron Devices*, vol. 67, no. 7, pp. 3010–3013, Jul. 2020.

**Gaurang Bansal** (Member, IEEE) received the bachelor's and master's degrees from BITS Pilani, Pilani, India, in 2018 and 2020, respectively. He is currently pursuing the Doctoral degree with National University of Singapore (NUS), Singapore, under Prof. B. Sikdar.

His research experience includes mentorship with the IoT lab, BITS Pilani from 2017 to 2020, with Dr. V. Chamola. He is currently working on "Security and Privacy Solutions for Unmanned Aerial Vehicles Networks." His works include the use of cryptography, deep learning, and network security techniques. He has more than authored 30 publications in top-tier conferences and journals, such as IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE INFOCOM, and IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING.

Dr. Bansal is a recipient of the Google Ph.D. Fellowship and NUS Presidential Fellowship.

**Biplab Sikdar** (Senior Member, IEEE) received the B.Tech. degree in electronics and communication engineering from North Eastern Hill University, Shillong, India, in 1996, the M.Tech. degree in electrical engineering from Indian Institute of Technology Kanpur, Kanpur, India, in 1998, and the Ph.D. degree in electrical engineering from Rensselaer Polytechnic Institute, Troy, NY, USA, in 2001.

He was a Faculty with Rensselaer Polytechnic Institute from 2001 to 2013, first as an Assistant Professor and then as an Associate Professor. He is currently a Professor and the Head of Department with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, where he serves as the Area Director of Communications and Networks Lab. His current research interests include wireless network, and security for Internet of Things and cyber–physical systems.

Dr. Sikdar served as an Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS from 2007 to 2012. He currently serves as an Associate Editor for the IEEE TRANSACTIONS ON MOBILE COMPUTING. He has served as a TPC in various conferences, such as IEEE LANMAN, GLOBECOM, BROADNETS, and ICC.