

Developing Secure Hardware for UAV Authorisation Using Lightweight Authentication

Muhammet A. Sen

School of Aerospace, Transport
and Manufacturing(SATM)
Cranfield University, United Kingdom
muhammet.sen@cranfield.ac.uk

Saba Al-Rubaye

School of Aerospace, Transport
and Manufacturing(SATM)
Cranfield University, United Kingdom
s.alrubaye@cranfield.ac.uk

Antonios Tsourdos

School of Aerospace, Transport
and Manufacturing(SATM)
Cranfield University, United Kingdom
a.tsourdos@cranfield.ac.uk

Abstract— In this research, a unique authorisation strategy for unmanned aerial vehicle (UAV) systems is presented. UAVs are being used more frequently in a variety of military and civilian missions, including observation, reconnaissance, transportation, search and rescue, traffic surveillance, weather forecasting, and logistics operations. To address the need for secure authorization in UAV networks, we propose an innovative scheme that leverages Physically Unclonable Functions (PUFs) within the Flying Ad-Hoc Network (FANET) structure. Our plan strengthens the security and integrity of UAV operations by incorporating PUF technology. The suggested authorization architecture is appropriate for the dynamic and resource-constrained FANET context because it ensures effective and trustworthy authentication while minimising computational cost. Through this study, we hope to further the creation of reliable permission systems for UAV systems used in both military and civilian applications.

Keywords—Hardware Security, PUF, FANET, UAV, Authorisation, Cyber Attacks

I. INTRODUCTION

Unmanned Aerial Vehicle (UAV) technology has a great importance in the military field, as well as achieved serious success in the civilian field and expanded its field of use. It is crucial for military operations, particularly in those involving reconnaissance, surveillance, and target acquisition, electronic warfare, communications relay, search and rescue operations, as well as the distribution of supplies, equipment, weapons and munitions to troops in the field or combat [1]. In addition to being utilised in military application, it is also employed in civilian applications including search and rescue missions, traffic surveillance, weather forecasting, and the delivery of goods and medical supplies. Drone swarms are an additional idea that has developed along with UAV technology. A drone swarm is formed when multiple drones come together and work in coordination to achieve a common goal [2]. Drone swarm networks are primarily concerned with organising the activities of a group of drones, whereas FANET networks are concerned with employing drones to build a communication infrastructure [3]. In this paper, it is offered a lightweight and secure authentication method for UAVs in FANET structure.

Unmanned Aerial System and Drone Swarm

In this section unmanned aerial system (UAS) components are introduced, communication architecture types are identified and examples of how UAVs cooperate in FANETs are provided. Unmanned Aerial Systems are composed of five primary components [2], [3];

1- UAV

2- UAV Operating System

3- Ground Control Station (GCS)

4- Payloads

5- Repair and Support System

The communication link between UAV and GCS is significant in terms of instant data exchange and security. There are two primary communication topologies for UAVs regarding of data transmission: centralised and decentralised.

Centralised architectures, includes UAV to GCS, UAV to satellite, and UAV to cellular network. The UAV establishes direct communication with a central control point. This also results in the system's vulnerability to single points of failure. In decentralised architecture, communication between the UAV and the GCS can occur through direct or indirect means. Additionally, within the system, a special UAV known as the backbone UAV plays a crucial role as a bridge connecting the GCS and other UAVs within the network. This backbone UAV facilitates communication and coordination among the various UAVs in the decentralized architecture. Table I demonstrates five different connection architectures used in UAV networks.

TABLE I. UAV Control Architectures

Type	Drone Swarm Networks and Key Characteristic			
	Description	Control	Decision Making	Resilience
Centralized	A central control system coordinates the actions of the drones	Centralized	Centralized	Low
Decentralized	Drones make decisions and coordinate actions without a central control system	Decentralized	Decentralized	High
Hierarchical	Drones are organized into a hierarchical structure with leaders and followers	Centralized	Hierarchical	Moderate
Hybrid	Combination of different types of networks, adapt to different scenarios	Hybrid	Hybrid	Depends on the scenario
Virtual	Simulation of a real drone swarm network, used for testing and training	Varies	Varies	N/A

A. Flying Ad-Hoc Networks

FANET stands for Flying Ad-hoc Network, which is a wireless communication system that allows drones to exchange data and communicate with each other. FANET is a type of Mobile Ad-hoc Network (MANET), which is a network of mobile devices that can establish a connection without requiring a fixed infrastructure [2], [4], [5]. FANET technology has received considerable attention in recent years due to the increasing use of drones in a number of industries. FANET enables drones to collaborate in a coordinated manner, which can be especially useful in

situations in which the drones must cover a large area or operate in challenging environments [6].

The main benefit of FANET lies in its ability to facilitate real-time communication and information exchange among drones, thereby enhancing operational efficiency and effectiveness. Furthermore, FANET enables drones to share crucial data such as their position, velocity, and other relevant information, leading to improved safety and reliability during flights [7], [8]. FANET networks can adopt either a centralized or decentralized structure. In a centralized FANET, a central controller manages communication between drones. Conversely, a decentralized FANET lacks a central controller, with each drone directly communicating with its neighboring drones.

Security Flaws

The use of unmanned aerial vehicles (UAVs), or drones, has become increasingly popular for both commercial and military applications. However, UAVs are vulnerable to cyber-attacks, which can compromise their mission, endanger people, and cause damage [9], [10]. The authentication phase holds significant importance in mitigating various attacks, including Man-in-the-Middle (MITM), node tampering, eavesdropping, replay attacks, masquerade attacks, and cloning attacks. By effectively configuring authentication protocols, these potential threats can be prevented, and the overall security of the system can be enhanced.

II. INTRODUCTION

The utilisation of FANET has gained noteworthy consideration in diverse domains such as military reconnaissance, calamity administration, and aerial imaging in recent times. In this section, an overview of the existing literature and research on the utilization of PUFs and other methods for authorization in FANETs is presented. To provide a thorough grasp of the state-of-the-art in this field, the literature on UAV-UAV and UAV-GCS authorisation is also investigated.

It is observed that in various recent research, Physically Unclonable Functions (PUF) are highly promising for supplying protection in IoT and embedded hardware applications [11]–[14]. These electronic devices give their products unclonable and distinctive identities because of the inherent randomness added throughout the manufacturing process. As a result, they can be used as hardware security primitives as well as hardware roots of trust. PUFs work via a challenge-response pair method, meaning that when a stimulus known as a challenge is delivered to them, the device produces an output known as a response [15]. Given the same challenge, it is expected that two separate PUF devices made using the same manufacturing method and arrangement will respond in different ways. The aforementioned PUF attribute is used in the proposed protocol to create a special session key for each device that has a PUF incorporated in it. For each UAV in the network, the implanted PUF chip would serve as a distinct, irreproducible fingerprint. Its behaviour is comparable to that of a function that accepts a challenge as an input and outputs a response in response.

$$R = \text{PUF}(C)$$

In the equation presented above, the variable C represents the challenge, and the variable R represents the response; both C and R are binary strings.

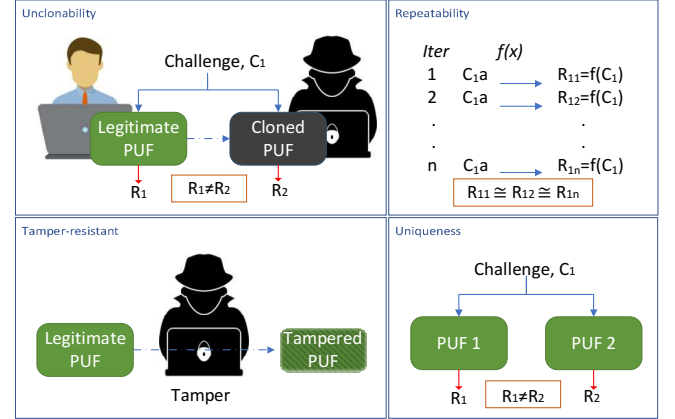


Fig. 1. Fundamental Properties of PUF

PUFs have attracted considerable attention for their potential to provide secure and lightweight authentication in various networked systems. Researchers have identified the potential of PUF-based approaches for authorising UAVs within FANETs. PUFs provide a way to give UAVs distinct, immutable identities that permit secure communication and teamwork between authorised nodes.

A multi-layer defence strategy against buffer overflows, and Denial of Service (DoS) attacks on the Address Resolution Protocol (ARP) cache, has been presented by Hooper et al. in [16]. A robust adversarial architecture that can carry out fault injection, side-channel, and physical attacks on a UAV has been taken into consideration by Blazy et al. [17], [18]. They employ a keyed hash function to produce following keys from a single initial key that is stored in their scheme. To avoid an attack, they suggest using several streams of these key sequences. However, if a strong attacker manages to obtain just one key from each stream, they could simply keep track of all the keystreams at once, making any future communication insecure.

A simple security plan for surveillance UAV networks has been given by Abdallah et al. in [19], although it only ensures unilateral authentication, not reciprocal authentication. A system for certificateless group key authentication that is aimed at untrusted UAV networks is another piece of work [20]. This work is not lightweight because it relies on elliptical curve cryptography (ECC) and bilinear pairing. A short certificate-based signature distribution strategy is proposed in another article [21], although it is equally complicated because it makes use of public key cryptography methods.

Recent studies [22]–[24] have demonstrated that PUFs are particularly successful in securing wireless and unmanned IoT environments. In [25], Chen and Willems offer a PUF-based secret key generation system. The challenge-response pair of the PUF is used as an initialization condition for a chaotic system in a recent paper on UAV authentication utilising PUFs [26], to covertly exchange random seeds and produce a session key. Table II provides a summarized comparison of the features found in similar studies.

This study proposes an innovative method for session key creation using a PUF-based authorization mechanism and adding various hash functions to increase its complexity. It's important to avoid keeping secret keys and other important data in the UAV memory. Our protocol allows for the immediate creation of sensitive data by making use of the built-in PUF of the in-flight drone. The suggested protocol makes sure that the UAV node and the base station can mutually authenticate each other by making use of this generated data.

TABLE II. Comparison of Related Researches

P	Method	Cases	Verification	Advantages	Drawbacks
[11]	Ideal PUF Hash Function	UAV - UAV -GS	A) Formal Proof: Mao Boyd Logic B) Cryptanalysis (C1 to C12, D.Wang)	U2G and U2U authentication Session key agreement	It uses raw response Single point of failure
[12]	Noisy PUF Fuzzy Extractor	UAV - UAV -GS1 UAV -GS2	1)GNY Logic (extension of BAN logic) 2)Design Goals	Cross-domain authentication Session key agreement	Higher computational cost than ideal PUF Cases
[27]	Duffing Map	UAV - UAV	1)Chaotic System 2)Simulation on Omnet++	Exchange random seeds Using Chaotic System Session key agreement	Lack of user anonymity
[28]	Hash functions Fuzzy Extractor		NS2 simulation	Biometric verification IoD environment	Can not cope with tampering attack
[29]	Hash function		Formal Security Verification: AVISPA Scyther simulator	Session key agreement	
[30]	Ideal Puf Symmetric Key Encryption	UAV -GS	Mao-Boyd Logic	Dynamic topology	It uses raw response. Centralized authentication

III. SYSTEM AND THREAT MODEL

The main objective of our system are to establish the session key and to accomplish mutual authentication among Ground Control Stations and UAVs. As seen in Fig. 2, the system model under consideration in this work consists of authorised UAVs and a ground station. Compared to the ground station, UAVs are less capable of memory and computation.

This system allows for the connection of many UAVs to a single ground station. This study aims to execute mutual authentication between two entities in order to establish a secure communication session between a UAV U_i and the ground control station G_i . Additionally, it will be demonstrated that any two UAVs, namely U_1 and U_2 , can

establish a secure session by expanding the same mutual authentication technique.

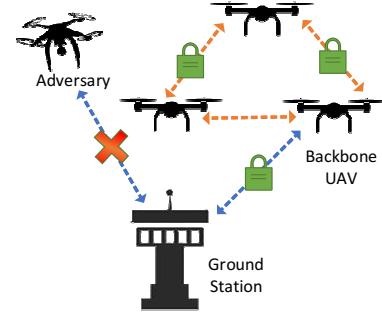


Fig. 2. System Structure

The system model illustrates the scenario within a classical FANET architecture, where communication is facilitated through the backbone UAV. Prior to delving into the details of the proposed protocol, it is essential to define the terms utilized in the protocol. Table III presents the definitions of these terms for better understanding and clarity.

TABLE III. The List of Notations

Notation	Description
U_i	The ID of UAV
Tid_i	Temporary ID of the UAV
N_{UX}	Random Nonce
C_x	x^{th} challenge parameter of PUF
R_x	x^{th} response parameter of PUF
G_i	The ID of Ground Station
GCS	Ground Control Station
$h()$	Hash Function
$\oplus$	XOR Operation
$ $	Concatenation
Sk_i	Session Key

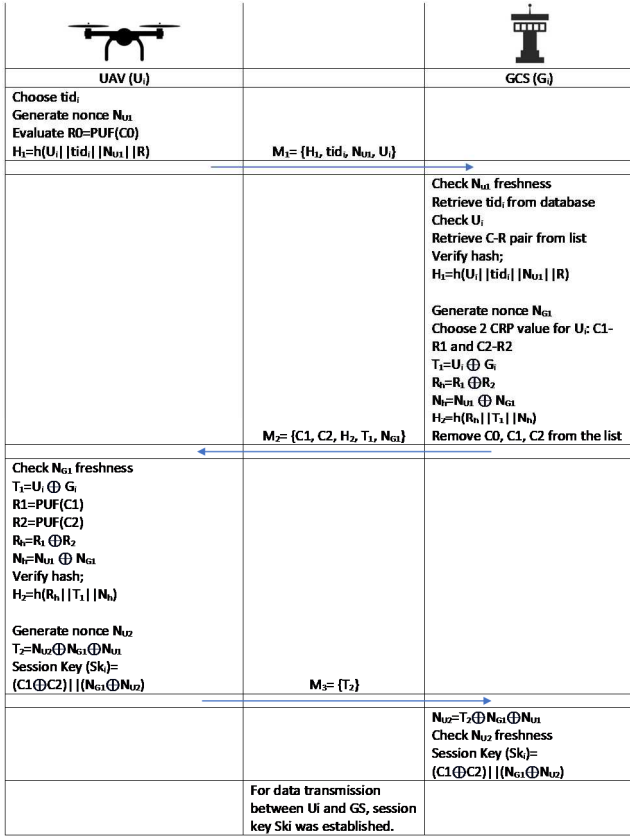


Fig. 3. The authentication phase

IV. PROPOSED AUTHENTICATION PROTOCOL

The suggested approach is mainly divided into three phases. The three phases of the proposed approach are the UAV registration phase, the UAV-GS authentication phase, and the UAV-UAV authentication phase.

To preserve the UAV's genuine identity information while the authentication process is being started and finished, the system has adopted disposable pseudonyms and mask identities.

A. Registration Phase

1) Before deployment, each UAV U_i registers with the GS at the ground station.

2) During registration, the GS stores in its database a list of challenge-response pairings (C, R) generated by U_i's PUF.

3) Each UAV U_i is assigned a temporary identifier tid_i by GS, and GS also stores the UAV's permanent identity (U_i).

4) In the GS, the G_i, tid, C-R Pair list, and U_i are stored, whereas the UAV holds C0, tid, G_i, and U_i.

B. Authentication Phase

The proposed scheme's authentication and key negotiation procedure is described in detail below. Fig. 4 illustrates the proposed scheme's computations conducted on a valid UAV U_i and a GCS G_i, as well as the communications exchanged between them. Through this protocol, it is made sure that only a valid UAV can

authenticate with GS before starting a secure communication session. A secure session key is created between UAV and GS during the authentication procedure, which can be utilised for future communications.

1) A UAV U_i creates the response R utilising the challenge C that has been previously stored, when it has to authenticate with GCS.

2) It sends the message M₁ = {H₁, tid_i, N_{U1}} to GCS. Its temporary ID tid_i, a random nonce N_{U1}, and a hash calculated a H₁=h(U_i||tid_i||N_{U1}||R).

3) GCS queries its database for any entry corresponding to the received ID tid_i. Furthermore, it determines the state of freshness of N_{U1} (It can't be the same as the N_{U1} of the earlier authentications). After both of these requirements are met, U_i's authentication request will be processed, otherwise the request will be ignored. After confirming the hash, GS searches its database for the relevant challenge-response pair (C, R). A nonce N_{G1} is produced, 2 unused C-R pairs are chosen from database, and the actions listed below are carried out.

$$T_1 = U_i \oplus G_i$$

$$R_h = R_1 \oplus R_2$$

$$N_h = N_{U1} \oplus N_{G1}$$

$$H_2 = h(R_h || T_1 || N_h)$$

$$M_2 = \{C_1, C_2, H_2, T_1, N_{G1}\}$$

4) GCS then sends M₂ = {C₁, C₂, H₂, T₁, N_{G1}} to U_i.

5) UAV U_i check the freshness of N_{G1} and obtain G_i using T₁ in the incoming message and its own ID U_i.

$$G_i = U_i \oplus T_1$$

$$R_1 = \text{PUF}(C_1)$$

$$R_2 = \text{PUF}(C_2)$$

$$R_1 \oplus R_2 = R_h$$

$$N_{U1} \oplus N_{G1} = N_h$$

Using C₁ and C₂ challenges as input, the PUF on the UAV U_i creates R₁ and R₂ responses. These responses go through bitwise xor to produce the R_h value. Utilising N_{U1}, the initial nonce it generated, and N_{G1}, the nonce from GCS, it is able to derive N_h. The validity of the hash generated by GCS is then verified. The UAV will terminate the authentication process if the hash validation fails. Upon successful verification, a random nonce N_{U2}, is generated. As shown below, T₂ is encoded with this newly created data.

$$T_2 = N_{U2} \oplus N_{G1} \oplus N_{U1}$$

The computation of the session key Sk_i, which will be used for subsequent communication, is displayed below.

$$Sk_i = (C_1 \oplus C_2) || (N_{G1} \oplus N_{U2})$$

6) The M₃ = {T₂} message is finally transmitted by the UAV to the GCS. The GCS calculates N_{U2} using the following formula after receiving the T₂ message and check its freshness. After that, the Sk_i is computed as in the UAV.

$$N_{U2} = T_2 \oplus N_{G1} \oplus N_{U1}$$

$$Sk_i = (C_1 \oplus C_2) || (N_{G1} \oplus N_{U2})$$

8) Consequently, the mutual authentication process between the UAV and the GCS has been successfully executed. Subsequently, any additional communication that takes place is safeguarded through encryption, which is facilitated by the session key Sk_i that has been established.

9) As seen in the equation below, in order to maintain user anonymity, both the UAV and the GCS create a fresh temporary id, tid' for the UAV, which will be utilised during the subsequent authentication.

$$tid' = H(N_{G1} \parallel tid_i \parallel R_h)$$

V. SECURITY ANALYSIS

In order to assess the security of the proposed authentication phase between the UAV and GCS, a formal security proof is presented in Section V-A, along with traditional cryptanalysis in Section V-B. The adoption of such a methodology is deemed imperative as it ensures the attainment of both security and adaptability, given that relying solely on formal proof is deemed inadequate according to sources [31], [32]. The Mao Boyd logic [33] has been employed to carry out formal verification of the reliability of our protocol. Additionally, our cryptanalysis approach is based on the 12 evaluation criteria (C1 - C12) utilised by the authors in [35].

A. Formal Proof

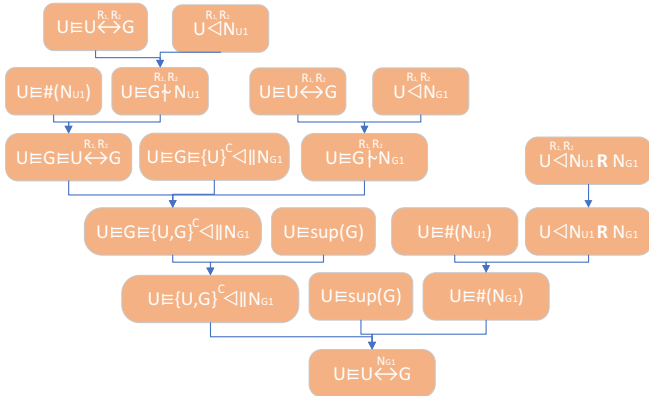


Fig. 4. Logical Proof of the protocol

To establish the validity of the statement "U believes $NG1$ is a good secret between U and G," we present a proof in Mao Boyd logic. The statements forming this proof are expressed through equations (i) - (xv), while a visual representation is depicted in Fig. 4. This proof demonstrates the logical justification for the assertion, supporting the understanding that U holds $NG1$ as a secure secret shared exclusively between U and G.

The responses $R1$ and $R2$ corresponding to the challenge-response pairs (C1, $R1$) and (C2, $R2$) of U are stored in G, therefore "U believes $R1$ and $R2$ are a good secret between U and G" (i). Using $R1$ and $R2$, the UAV U is able to decipher Rh and $H2$ in the second message of the protocol $M_2 = \{C_1, C_2, H_2, T_i, N_{G1}\}$ to get $NU1$ and $NG1$. Therefore, "U sees $NU1$ with decipher keys $R1$ and $R2$ " (ii) and "U sees $NG1$ with decipher key $R1$ and $R2$ " (iii).

$$U \equiv U \leftrightarrow G^{R1, R2} \quad (i)$$

$$U \equiv N_{U1} < N_{U1}^{R1, R2} \quad (ii)$$

$$U \equiv N_{G1} < N_{G1}^{R1, R2} \quad (iii)$$

By applying the authentication rule to statements (i) and (ii), we can derive the statement "U believes that G encrypted $NU1$ using $R1$ and $R2$ " (iv). Similarly, by applying the same rule to statements (i) and (iii), we obtain the statement "U believes that G encrypted $NG1$ using $R1$ and $R2$ " (v). Considering that U generates a new nonce NG each time, we can conclude that "U believes that $NU1$ is fresh" (vi).

Subsequently, we apply the nonce-verification rule to statements (iv) and (vi), leading to the conclusion "U believes that G believes that $R1$ and $R2$ is a good secret between U and G" (vii). This logical deduction confirms the belief held by U that G considers $R1$ and $R2$ to be a secure secret shared exclusively between U and G.

$$U \equiv G \vdash N_{U1}^{R1, R2} \quad (iv)$$

$$U \equiv G \vdash N_{G1}^{R1, R2} \quad (v)$$

$$U \equiv \#(N_{U1}) \quad (vi)$$

$$U \equiv G \equiv U \leftrightarrow G^{R1, R2} \quad (vii)$$

Given that G consistently generates a new nonce NB with each iteration, U recognizes that $NG1$ remains undisclosed to anyone other than G. Therefore, U holds the belief that "G believes that no one other than U has access to $NG1$ " (viii). By applying the confidentiality rule to statements (v), (vii), and (viii), we can deduce that "U believes that G believes that no one other than U and G has access to $NG1$ " (ix). This inference reinforces the understanding that U and G are the sole entities with authorized access to $NG1$, as per U's beliefs and G's acknowledgment.

$$U \equiv G \equiv \{U, G\}^C < N_{G1} \quad (viii)$$

$$U \equiv G \equiv \{U, G\}^C < N_{G1} \quad (ix)$$

The protocol assumes that U holds the belief that the ground station G is secure and trusted, or in other words, "U believes that G is the super-principal" (x). By applying the super-principal rule to statements (ix) and (x), we can conclude that "U believes that no one other than U and G has access to $NG1$ " (xi). This indicates that U trusts that only U and G possess the necessary access privileges to $NG1$, reinforcing the security and confidentiality of the information exchanged between them.

$$U \equiv \text{sup}(G) \quad (x)$$

$$U \equiv \{U, G\}^C < N_{G1} \quad (xi)$$

We can identify $NU1$ as the challenge, while $NG1$ is recognized as the response. It is important to note that $NU1$ and $NG1$ are distinct from the challenge-response pair (C, R) associated with the PUF. Consequently, the statement "U can see the replied challenge $NU1$ and the response $NG1$ with decipher keys $R1$ and $R2$ " (xii) holds true. By applying the intuitive rule to statement (xii), we can simplify it to "U

can see the replied challenge NU1 and the response NG1" (xiii).

$$U \xleftarrow{R_1, R_2} N_{U1} R_{G1} \quad (xii)$$

$$U \xleftarrow{R_1, R_2} N_{U1} R_{G1} \quad (xiii)$$

By applying the fresh rule to statements (vi) and (xiii), we can conclude that "U believes NG1 is fresh" (xiv). Next, by applying the good-key rule to statements (x), (xi), and (xiv), we have successfully proven the statement "U believes that NG1 is a good secret known only to U and G" (xv). This logical deduction affirms U's belief that NG1 remains confidential and is securely known only to U and G.

$$U \models \#(N_{G1}) \quad (xiv)$$

$$U \equiv U \xleftrightarrow{N_{G1}} G \quad (xv)$$

In summary, it has been demonstrated that the statement "G believes that NG1 is a good secret known only to U and G" can be proven using a similar logical approach.

Furthermore, a similar reasoning can be applied to prove that adversaries cannot gain access to NU2, R1, and R2. The secrecy of NG1, NU2, R1, and R2 holds regardless of the type of attack employed by adversaries, be it a masquerade attack, a man-in-the-middle attack, or a replay attack. As a result, the communicated data remains incomprehensible and protected from unauthorized access by adversaries.

B. Cryptanalysis

This evaluation depends on the authors' list of security requirements from [35], which eliminates uncertainties and redundancies that were common in the field of security protocols. The examination of the criteria C1–C4–C6–and C9 has been excluded because they entail smart cards and user-input passwords, neither of which applicable to the protocol offered in this paper. This cutting-edge security evaluation technique has already been incorporated into a number of papers, including [36], [37]. In this Cryptanalysis, G, U, and A represent the GCS, the UAV, and the Adversary, respectively.

[C5] Resistance to known attacks: The suggested architecture is resistant to the following attacks: man-in-the-middle (MITM) (C5.1), masquerade (C5.2), replay (C5.3), node tampering (C5.4), cloning (C5.5), and de-synchronization (C5.6). In our study, in order to prevent MITM attack, (C, R) pairs registered in G upon registration will be considered as one-time use. An adversary who lies between the U and the G in our system may be able to obtain the challenge-response pair provided during the last authentication. If G sends the same 'challenge' during a subsequent authentication process, the adversary can transmit the 'response' to that 'challenge' to G without sending the G's authentication message to the U. A previously utilised 'challenge' value will not be transmitted back to the U for another authorisation. This means that using a new pair from the CRP list will provide security against MITM attacks. The adversary A is incapable of masquerade a UAV due to the absence of an identical PUF. Additionally, it is unable to mimic a G as it lacks the appropriate (C, R) pairs. The proposed protocol ensures that any attempts made by A to replay previous messages to GCS or UAV will be unsuccessful. This is due to the

utilisation of a new (C, R) pair for authentication purposes in each session, which prevents A from obtaining access to UAVs (C, R) pair. Due to the inherent unclonability of PUFs, it is not possible for party A to successfully clone the UAV. As a result, the scheme is deemed safe for cloning attacks. The suggested technique is safe for node tampering attacks even if A successfully captures the UAV since any effort to tamper on the PUF will turn the PUF useless. Therefore, even in the case of a device capture, A cannot engage in malicious activities like node tampering; in addition, no crucial information linked to the authentication process is saved in Ui.

[C7] Provision of key agreement: Under the heading Formal Proof in Section V, the confidentiality of Nu1, Ng1, and Nu2 has been established. Additionally, "Rh" is created by xoring two different "response". Authentication is achieved in each session through the use of new (C, R) pairs, resulting in the utilisation of distinct Rs for every session. The proposed protocol establishes a fresh and secure session key denoted as $Sk_i = (C_1 \oplus C_2) \parallel (N_{G1} \oplus N_{U2})$ following a successful authentication.

[C8] No clock synchronization: The freshness of messages is ensured in our protocol through the utilisation of random nonces, rather than timestamps. The suggested security scheme is devoid of issues pertaining to both clock synchronisation and time delay. Specifically, it obviates the need for GCS and UAV to synchronise their respective clocks.

[C10] Mutual authentication: As part of the suggested plan, GS is used to temporarily house a single PUF pair of UAVs before their deployment. This means that a session key can be formed between the UAV and the GS alone. Therefore, if a session key Sk_i is established, it signifies that mutual authentication has taken place, and that the verifying participants are valid.

[C11] User anonymity: Our protocol ensures user anonymity by updating Ui's temporary ID after every authentication using the formula $tid_i' = H(N_{G1} \parallel tid_i \parallel R_h)$ at the final stage of the process. Without transfer message to each other, Ui and GS can independently calculate the same temporary ID. For a particular challenge C, the temporary ID cannot be determined by anybody other than Ui or GS, who alone know the PUF response R.

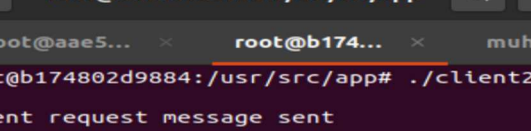
[C12] Forward secrecy: The suggested technique ensures that the security of subsequent sessions remains unharmed even if A is successful in guessing the present-session key Sk_i . This is due to the fact that A cannot get the response for the following session, R, which is necessary to construct the session key for the following session, as has already been established under the heading Formal Proof in Section V. A won't even be able to follow UAV thanks to the suggested protocol's promise of C11, much less endanger any upcoming sessions. Because of this, the suggested procedure ensures absolute forward secrecy. Due to its inability to collect the necessary secrets to produce them, A will also be unable to obtain the session key of any previous session. This protocol so also ensures perfect backward secrecy.

A. Security Comparison



Schemes	Communication Cost (bits)
[34]	1696
[26]	1952
[11]	1600
Ours	1408

Fig. 5 provides a visual representation of the communication cost in terms of total bit utilization during the authentication process. The proposed protocol's reliability has been validated against cyber-attacks, specifically Section V.A and V.B. The results demonstrate that the proposed protocol outperforms other comparative studies, offering a more efficient communication cost. This observation highlights the protocol's success in achieving a balance between security and communication efficiency, making it a favourable choice compared to existing approaches.



```
root@b174802d9884: /usr/src/app
root@aae5... x root@b174... x muhsen@
root@b174802d9884: /usr/src/app# ./client2
Client request message sent
Challenge message is received value is 193
response send from client is 194
Response Message sent
ACK packet received
Authentication Successfull
root@b174802d9884: /usr/src/app#
```

```
root@aae5de0fac2e: /usr/src/app# ./server2
request message received form client
request message msgType value: 0
This is the request packet
Challenge Message Sent
    challenge message value is : 193
Response Message Received
Response packet received
response value is 194
Authentication Successfull
ACK authentication successfull packet sent
root@aae5de0fac2e: /usr/src/app#
```

VII. CONCLUSION

The current protocol will be extended to include authentication between UAVs. Furthermore, the system will incorporate Error Correction Codes to mitigate errors in the

Responses generated by the PUF under real-world circumstances.

REFERENCES

- [1] J. Yaacoub, H. Noura, O. Salman, and A. Chehab, "Security analysis of drones systems: Attacks, limitations, and recommendations," no. January, 2020.
- [2] K. Y. Tsao, T. Girdler, and V. G. Vassilakis, "A survey of cyber security threats and solutions for UAV communications and flying ad-hoc networks," *Ad Hoc Networks*, vol. 133, no. May, 2022, doi: 10.1016/j.adhoc.2022.102894.
- [3] A. Tsourdos, B. White, and M. Shanmugavel, *Cooperative path planning of unmanned aerial vehicles*. Wiley, 2010. doi: 10.1002/9780470974636.
- [4] I. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying Ad-Hoc Networks (FANETs): A survey," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013, doi: 10.1016/j.adhoc.2012.12.004.
- [5] A. Bujari, C. T. Calafate, J. C. Cano, P. Manzoni, C. E. Palazzi, and D. Ronzani, "Flying ad-hoc network application scenarios and mobility models," *Int. J. Distrib. Sens. Networks*, vol. 13, no. 10, pp. 1–17, 2017, doi: 10.1177/1550147717738192.
- [6] J. Condomines, R. Zhang, and N. Larrieu, "Network Intrusion Detection System for UAV Ad-hoc Communication," pp. 1–34, 2012.
- [7] I. Bekmezci, E. Senturk, and T. Turker, "Security Issues in Flying Ad-Hoc Networks (FANETs)," *Journal of Aeronautics and Space Technologies*, vol. 9, no. October, pp. 13–21, 2015.
- [8] A. A. Deshpande, F. Chiariotti, and A. Zanella, "SMURF: Reliable Multipath Routing in Flying Ad-Hoc Networks," *2020 Mediterr. Commun. Comput. Netw. Conf. MedComNet 2020*, 2020, doi: 10.1109/MedComNet49392.2020.9191526.
- [9] A. Alooseel, S. Al-Rubaye, A. Zolotas, and C. Shaw, "Attack-Detection Architectural Framework Based on Anomalous Patterns of System Performance and Resource Utilization - Part II," *IEEE Access*, vol. 9, pp. 87611–87629, 2021, doi: 10.1109/ACCESS.2021.3088411.
- [10] A. Alooseel, H. He, C. Shaw, and M. A. Khan, "Analytical Review of Cybersecurity for Embedded Systems," *IEEE Access*, vol. 9, pp. 961–982, 2021, doi: 10.1109/ACCESS.2020.3045972.
- [11] 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, 2020, doi: 10.1109/TVT.2020.3033060.
- [12] C. Tian, Q. Jiang, T. Li, J. Zhang, N. Xi, and J. Ma, "Reliable PUF-based mutual authentication protocol for UAVs towards multi-domain environment," *Comput. Networks*, vol. 218, no. September, p. 109421, 2022, doi: 10.1016/j.comnet.2022.109421.
- [13] C. Pu, A. Wall, K. K. R. Choo, I. Ahmed, and S. Lim, "A Lightweight and Privacy-Preserving Mutual Authentication and Key Agreement Protocol for Internet of Drones Environment," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9918–9933, 2022, doi: 10.1109/JIOT.2022.3163367.
- [14] L. Zhang, J. Xu, M. S. Obaidat, X. Li, and P. Vijayakumar, "A PUF-based lightweight authentication and key agreement protocol for smart UAV networks," *IET Commun.*, vol. 16, no. 10, pp. 1142–1159, 2022, doi: 10.1049/cmu2.12295.
- [15] T. Güneysu, H. H. Eds, and D. Hutchison, *CHES 2015 Proceedings*. 2015.
- [16] M. Hooper *et al.*, "Securing commercial WiFi-based UAVs from common security attacks," in *MILCOM 2016 - 2016 IEEE Military Communications Conference*, 2016, pp. 1213–1218. doi: 10.1109/MILCOM.2016.7795496.
- [17] O. Blazy *et al.*, "An Efficient Protocol for UAS Security," in *2017 Integrated Communications, Navigation and Surveillance Conference (ICNS)*, 2017, pp. 1–21. doi: 10.1109/ICNSURV.2017.8011987.
- [18] T. Alladi, V. Chamola, B. Sikdar, and K.-K. R. Choo, "Consumer IoT: Security Vulnerability Case Studies and Solutions," *IEEE Consum. Electron. Mag.*, vol. 9, no. 2, pp. 17–25, 2020, doi: 10.1109/MCE.2019.2953740.
- [19] A. Abdallah, M. Z. Ali, J. Mišić, and V. B. Mišić, "Efficient Security Scheme for Disaster Surveillance UAV Communication Networks," *Information*, vol. 10, no. 2, 2019, doi: 10.3390/info10020043.
- [20] B. Semal, K. Markantonakis, and R. N. Akram, "A Certificateless Group Authenticated Key Agreement Protocol for Secure Communication in Untrusted UAV Networks," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, 2018, pp. 1–8. doi: 10.1109/DASC.2018.8569730.
- [21] G. K. Verma, B. 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, 2020, doi: 10.1109/JSYST.2019.2921798.
- [22] B. Chatterjee, D. Das, S. Maity, and S. Sen, "RF-PUF: Enhancing IoT Security Through Authentication of Wireless Nodes Using In-Situ Machine Learning," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 388–398, 2019, doi: 10.1109/JIOT.2018.2849324.
- [23] D. Choi, S.-H. Seo, Y.-S. Oh, and Y. Kang, "Two-Factor Fuzzy Commitment for Unmanned IoT Devices Security," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 335–348, 2019, doi: 10.1109/JIOT.2018.2837751.
- [24] T. Alladi, V. Chamola, and S. Zeadally, "Industrial Control Systems: Cyberattack trends and countermeasures," *Comput. Commun.*, vol. 155, pp. 1–8, 2020, doi: https://doi.org/10.1016/j.comcom.2020.03.007.
- [25] B. Chen and F. M. J. Willems, "Secret Key Generation Over Biased Physical Unclonable Functions With Polar Codes," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 435–445, 2019, doi: 10.1109/JIOT.2018.2864594.
- [26] C. Pu and Y. Li, "Lightweight Authentication Protocol for Unmanned Aerial Vehicles Using Physical Unclonable Function and Chaotic System," in *2020 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*, 2020, pp. 1–6. doi: 10.1109/LANMAN49260.2020.9153239.
- [27] C. Pu and Y. Li, "Lightweight Authentication Protocol for Unmanned Aerial Vehicles Using Physical Unclonable Function and Chaotic System," *IEEE Work. Local Metrop. Area Networks*, vol. 2020-July, 2020, doi: 10.1109/LANMAN49260.2020.9153239.
- [28] M. Wazid, A. K. Das, N. Kumar, A. V. Vasilakos, and J. J. P. C. 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, 2019, doi: 10.1109/JIOT.2018.2888821.
- [29] P. Mall, R. Amin, M. S. Obaidat, and K. F. Hsiao, "CoMSeC++:

- PUF-based secured light-weight mutual authentication protocol for Drone-enabled WSN,” *Comput. Networks*, vol. 199, no. November 2020, p. 108476, 2021, doi: 10.1016/j.comnet.2021.108476.
- [30] G. Bansal and B. Sikdar, “S-maps: Scalable mutual authentication protocol for dynamic uav swarms,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 11, pp. 12088–12100, 2021, doi: 10.1109/TVT.2021.3116163.
- [31] D. Wang, D. He, P. Wang, and C.-H. Chu, “Anonymous Two-Factor Authentication in Distributed Systems: Certain Goals Are Beyond Attainment,” *IEEE Trans. Dependable Secur. Comput.*, vol. 12, no. 4, pp. 428–442, 2015, doi: 10.1109/TDSC.2014.2355850.
- [32] D. Wang, H. Cheng, D. He, and P. Wang, “On the Challenges in Designing Identity-Based Privacy-Preserving Authentication Schemes for Mobile Devices,” *IEEE Syst. J.*, vol. 12, no. 1, pp. 916–925, 2018, doi: 10.1109/JSYST.2016.2585681.
- [33] W. Mao and C. Boyd, “Towards formal analysis of security protocols,” in *[1993] Proceedings Computer Security Foundations Workshop VI*, 1993, pp. 147–158. doi: 10.1109/CSFW.1993.246631.
- [34] W. Mao and C. Boyd, “Towards formal analysis of security protocols,” *Proc. Comput. Secur. Found. Work.*, pp. 147–158, 1993, doi: 10.1109/CSFW.1993.246631.
- [35] D. Wang and P. Wang, “Two Birds with One Stone: Two-Factor Authentication with Security beyond Conventional Bound,” *IEEE Trans. Dependable Secur. Comput.*, vol. 15, no. 4, pp. 708–722, 2018, doi: 10.1109/TDSC.2016.2605087.
- [36] D. Wang, W. Li, and P. Wang, “Measuring Two-Factor Authentication Schemes for Real-Time Data Access in Industrial Wireless Sensor Networks,” *IEEE Trans. Ind. Informatics*, vol. 14, no. 9, pp. 4081–4092, 2018, doi: 10.1109/TII.2018.2834351.
- [37] S. Roy, A. K. Das, S. Chatterjee, N. Kumar, S. Chattopadhyay, and J. J. P. C. Rodrigues, “Provably Secure Fine-Grained Data Access Control Over Multiple Cloud Servers in Mobile Cloud Computing Based Healthcare Applications,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 1, pp. 457–468, 2019, doi: 10.1109/TII.2018.2824815.



DSSPACE

<https://dspace.org/>

Developing secure hardware for UAV authorisation using lightweight authentication

Sen, Muhammet A.; Al-Rubaye, Saba; Tsourdos, Antonios

2023-11-10

Institute of Electrical and Electronics Engineers (IEEE)

<https://doi.org/10.1109/DASC58513.2023.10311211>

Sen MA, Al-Rubaye S, Tsourdos A. (2023) Developing secure hardware for UAV authorisation using lightweight authentication. In: 2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC), 1-5 October 2023, Barcelona, Spain