On the Security of the Novel Authentication Scheme for UAV-Ground Station and UAV-UAV Communication

Mustapha Benssalah[1] and Karim Drouiche[2]
[1]*Signal Processing Laboratory, Ecole Militaire Polytechnique, BP 17 Bordj El Bahri, 16046 Algiers, Algeria*
[2]*LIK Neuville Sur Oise, Cergy Pontoise University, Cergy-Pontoise CEDEX, 95000, France*

Keywords: IoD, Authentication, UAV, PUF, Security Analysis.

Abstract: With the unexpected increase in the number of commercialized and marketed UAVs in the last few years, both in the civilian and military fields, the security and privacy remain the exceedingly urgent problem of national security for many countries over the world. In fact, it is imperative that drone security and privacy issues have to be properly and utterly addressed by drone manufacturers as well as commercial operators, via implementing efficient authentication mechanisms executed between the system entities before any exchange of sensitive information. In this paper, we examine in depth the security of the PUF-based authentication scheme published most recently by Alladi et al. in one of the renowned international scientific journals "IEEE Transactions on Vehicular Technology". Our results indicate that the claimed security performance of this scheme has been overestimated. We show that Alladi et al.'s scheme is prone to the secret session key disclosure attack. We demonstrate that the attacker can easily reveal the shared secret and decrypt all the exchanged messages for both UAV-Ground Station (*GS*) and UAV-UAV authentication phases. To mitigate the revealed issues, some possible improvements are suggested for this scheme. Further, via formal security analysis, using Random Oracle, we show that Alladi *et al.*'s improved IoD scheme could deliver all the merits of the original scheme and can prevent the aforementioned vulnerabilities.

1 INTRODUCTION

Unmanned aerial vehicles (UAVs) are now widely used for both military and civilian applications, including package delivery, traffic surveillance, and search and rescue missions. UAV networks are rapidly evolving into the Internet of Drones, a layered network control design, with the help of embedded sensors and the acceptance of Internet of Things (IoT) as one of the main approaches in next generation (5G) (IoD) (Yahuza et al., 2021). IoD is referred to as a layered network control design that is primarily intended for managing UAV access to regulated airspace and offering navigation services between nodes. The Internet and other cutting-edge technologies like cloud computing, multi-access edge computing (MEC), artificial intelligence and communication networks enhance conventional UAV technology, creating enormous opportunities for future on-demand service-oriented and user-friendly IoD applications (Choudhary et al., 2018). Additionally, under the IoD concept, a large number of UAVs are grouped together to form a mesh network where each UAV, outfitted with sensors, gathers data from a specific airspace, disseminates/collects real-time data from other UAVs, and interacts with ground stations (Alsamhi et al., 2019). However, due to the highly sensitive nature of the collected data and the wireless nature of communication among the various entities that comprise the system, security and privacy of the exchanged information became a key concern (Lv, 2019). The pitfalls are to held responsible for security flaws, which result in significant loss of availability and resources, as well as a loss of privacy. Because the collected data in such a scenario is highly sensitive and decisive, so a secure and efficient authentication and key agreement (AKA) mechanism is required to ensure mutual authentication between the various entities uniting the system.

In recent years, the security and privacy in IoD systems have received a distinct attention. Numerous overviews and surveys on IoD security and privacy have been proposed in the literature (Lv, 2019; Choudhary et al., 2018; Lin et al., 2018; Alsamhi et al., 2019). Inspired by previous works that allow users to establish a shared key while being mutually

Benssalah, M. and Drouiche, K.
On the Security of the Novel Authentication Scheme for UAV-Ground Station and UAV-UAV Communication.
DOI: 10.5220/0012077400003555
In *Proceedings of the 20th International Conference on Security and Cryptography (SECRYPT 2023)*, pages 361-368
ISBN: 978-989-758-666-8; ISSN: 2184-7711

authenticated, many authentication schemes with innovative techniques in IoD environments have been proposed in the literature (Alladi et al., 2020b; Alladi et al., 2020a; Gope and Sikdar, 2020; Gope et al., 2021; Hussain et al., 2021). Let us concentrate on the most recent contributions concerning IoD security. Tian et al. (Tian et al., 2019) presented in 2019 an IoD privacy-preserving authentication scheme based on a digital signature scheme. Nonetheless, it is demonstrated that this scheme is insecure against location threats and physical attacks (Gope and Sikdar, 2020). TCALAS is a temporal credential anonymous lightweight authentication scheme for IoD proposed by Srinivas et al. in 2019. However, Ali et al. (Ali et al., 2020) showed that Srinivas et al. (Srinivas et al., 2019) scheme is susceptible to stolen verifier attacks and lacks anonymity. In 2020, Zhang et al. (Zhang et al., 2020) devised a lightweight AKA scheme based on bitwise XOR and one-way hash function operations to ensure mutual authentication between users and drones. Zhang et al. showed that their solution can resist to various known attacks and can achieve AKA-security under the random oracle model. Zhang et al. developed a lightweight AKA system based on one-way hash function and bitwise XOR operation to enable mutual authentication between users and drones in an IoD environment. (Zhang et al., 2020) demonstrated that their solution can withstand several known attacks and attain AKA-security under the random oracle paradigm. Furthermore, it provides improved functionality in terms of computation and transmission expenses. Nevertheless, Gope and Sikdar (Gope and Sikdar, 2020) examined Zhang et al's approach and proved its vulnerability to physical and forgery attacks. Furthermore, since the drone must store specific security credentials settings, it may be physically caught and all data stored in its memory accessed (Gope and Sikdar, 2020). Chen et al. suggested a traceable and privacy-preserving AKA for UAV communication control systems in 2020. In 2021, Yahuza et al. (Yahuza et al., 2021) showed that Chen et al. scheme is not secure under the widely used Canetti-Krawczyk (CK) adversary model. These attacks include the well-known session-specific temporary information attack, the partial key-escrow attack and the replay attack induced by a loss of integrity in the exchange messages. In 2021, Jan et al. (Jan et al., 2021) presented a lightweight message authentication scheme for IoD implementing hash function. (Jan et al., 2021) also showed that their protocol is immune to stolen-verifier and privileged insider attacks. On the other side, numerous notable PUF-based authentication solutions have been proposed in the literature in recent years with the goal of ensuring higher efficiency and degree of security generated by the Physically Unclonable Function (PUF) intrinsic properties such as unclonability, tamper-evident properties, and uniqueness. In this regard, Gope and Sikdar (Gope and Sikdar, 2020), Alladi et al., and Gope et al. (Gope et al., 2021) suggested efficient PUF-based authentication schemes for IoD environments.

In this paper, we first thoroughly examine the security of Alladi et al.'s (Alladi et al., 2020a) PUF-based authentication technique. Our findings suggest that this scheme's claimed security performance has been overestimated. As a result, we will show that Alladi et al.'s scheme is vulnerable to eavesdropping attack, in which an attacker who observes the insecure channel between the UAV and the ground station GS can easily extract the secret session key and then procure the overall exchanged communications between the various entities (UAV-UAV and UAV-GS). An improved scheme is suggested.

2 PRELIMINARIES

In this section, we will introduce some fundamental mathematical concepts that are used in the studied scheme.

2.1 Hash Function

A hash function is a one-way cryptographic function that transforms an entry string X of an arbitrary length to an output string Y. $X \in \{0,1\}^*$ into a condensed output string Y of specified length $Y \in \{0,1\}^n$, expressed as digests (Rogaway and Shrimpton, 2004). This one way function is expressed as $h(\cdot) : X \to Y$ and has the collision and pre-image resistances properties.

This characteristic can be described as follows: $Adv_{\hat{\mathcal{A}}}^{Hash}(t) = Pr[(x,x') \Leftarrow_R \hat{\mathcal{A}} : x \neq x' \text{ and } h(x) = h(x')]$, where $Pr[e]$ is the random occurrence e probability, $(x,x') \Leftarrow_R \hat{A}$ is the pair message (x,x') randomly picked by the attacker $\hat{\mathcal{A}}$ and $Adv_{\hat{\mathcal{A}}}^{Hash}(t)$ signifies the probability advantage gained over random picks by $\hat{\mathcal{A}}$ for a specified amount of time t. Then, if this function is collision-resistant, $Adv_{\hat{\mathcal{A}}}^{Hash}(t) < \varepsilon$ for small values of $\varepsilon > 0$.

2.2 Physically Unclonable Function

PUFs are used in the design of AKA systems as one of the most practical one way functions to provide a high security level against invasive and physical attacks. A PUF is defined as a pair of challenge/response pairs

(CRPs) for which the PUF generates R for a given input C such that $R = PUF(C)$. Otherwise, as reported in numerous contributions in the literature, a potential attacker can amass challenge response pairs (CRPs) from their build in PUF functions to establish a machine learning (ML) model that could be employed to predict the responses of future challenges with high accuracy (Shi et al., 2019; Yu and Wen, 2019).

Thus, we expect that in our enhanced authentication system, we would use new PUF functions similar to those described in (Wang et al., 2021) and (Wu et al., 2022) to avoid ML attacks (NoPUF and FLAM-PUF). The latter has novel countermeasures based on obfuscating challenge, which protects the PUF. For example, the prediction accuracy of modeling attacks over FLAM-PUF, which is basically composed of one Galois linear-feedback shift register (LFSR) and one Arbiter PUF (APUF), along with some simple logic gates, is around 50% under the commonly used ML techniques, namely support vector machines (SVMs), deep neural networks (DNNs), etc.

2.2.1 Definition

PUF_{Div}: $\{0,1\}^{L_1} \rightarrow \{0,1\}^{L_2}$ linked to a given thing `Div` is a function expressed with the following characteristics (Frikken et al., 2009):

1. PUF_{Div} is easy to calculate.
2. $Adv^{PUF}_{\hat{\mathcal{A}}}(L_2)$ is insignificant ($\leq \varepsilon$) in L_2 for any probabilistic polynomial-time adversary ($\hat{\mathcal{A}}$).
3. Bounded noise: in a wide range of circumstances, the distance between two given outputs from the PUF_{Div} on the same challenge C is at most d, i.e. $Pr[Dist_H(y,z) > d \mid y \leftarrow PUF_{Div_1}(C), z \leftarrow PUF_{Div_2}(C) \text{ and } C \leftarrow U_{L_2}] \leq \varepsilon$, for sufficiently small ε, where $Dist_H(\cdot,\cdot)$ is the Hamming distance.
4. Unique: the PUF_{Div} is specific to each technological equipment i.e. $Pr[Dist_H(y,z) \leq d \mid y \leftarrow PUF_{Div}(C), z \leftarrow PUF_{Div}(C) \text{ and } C \leftarrow U_{L_2}] \leq \varepsilon$, for a very tiny ε.

3 SECURITY ANALYSIS OF ALLADI et al. SCHEME

A PUF-based lightweight mutual authentication scheme was suggested by Alladi et al. (Alladi et al., 2020a) for the implementation of the Internet of Drones. This authentication scheme, known as SecAuthUAV, is recommended to secure communications between UAV-Ground station (GS) and UAV-UAV. SecAuthUAV was suggested to ensure a secure session between various entities without storing any sensitive data. In the event that the UAV is captured, this procedure will prevent the attacker from learning the secret keys that are kept in its memory. Furthermore, the authors argued that their scheme ensures crucial security aspects including mutual authentication, forward secrecy and UAV anonymity compared to recently proposed authentication schemes in this field. In addition, they showed that their scheme is resilient to a variety of well-known attacks, including the man-in-the-middle attack, masquerade attack, cloning attack, tampering attack, etc. However, in this section we will show how Alladi et al. scheme is susceptible to eavesdropping attack, in which an attacker might discover the shared secret and decode all the exchanged messages for both the UAV-Ground Station (*GS*) and UAV-UAV authentication phases. The main steps of this scheme are briefly described before we proceed to discuss the vulnerabilities that have been found.

3.1 Review of Alladi et al.'s Scheme

SecAuthUAV scheme consists of three phases, i.e. the UAV registration phase, UAV-GS authentication phase and the UAV-UAV authentication phase given in the following:

3.1.1 UAV Registration

- Before deployment, each UAV_{U_i} must always be enrolled with the *GS* using a secure channel.
- *GS* creates a temporary identity $TUID_i$ for each U_i and maintains the permanent identity *GID*.
- Utilizing U_i's PUF, a challenge-response pair (C,R) are produced and kept in the *GS* memory.
- The set $\{TUID_i, C, R\}$ is securely stored in the *GS*'s database (DB), while the set $\{TUID_i, GID, C\}$ is stored in the UAV's memory.

3.1.2 UAV-GS Authentication

During this phase, the UAV_{U_i} and the *GS* interact across an unsecured channel to establish mutual authentication and a session key for future interactions.

1. Once, a UAV_{U_i} needs to authenticate with *GS*, it computes the response $R = PUF(C)$ using the stored challenge C. Then, it generates a random nonce N_A and calculates $H(R \| TUID_i \| N_A)$ and it sends them together with its temporary identity $TUID_i$ to *GS* i.e. $M_1 = \{TUID_i, N_A, H(R \| TUID_i \| N_A)\}$.

2. Upon receiving M_1, GS checks the freshness of NA and requests its DB for any entry corresponding to the received $TUID_i$. If these conditions are not satisfied, the U_i's authentication demand will be rejected. Thereafter, once that hash value is checked, GS finds the corresponding challenge-response pair (C,R) from its DB. After that, it generates nonce N_B and subsequently splits R into two parts denoted here as K_1 and K_2, it calculates the message Q as follows:

$$X_1 = N_A \oplus K_2 \quad (1)$$

$$Y_2 = N_B \oplus X_1 \oplus K_1 \quad (2)$$

$$Q = (Y_2 \| X_1) \oplus (K_2 \| K_1) \quad (3)$$

3. GS broadcasts the message $M_2 = \{Q, H(Q \| GID \| N_A \| N_B)\}$ to U_i.
4. Upon the reception of M_2, it splits R into K_1 and K_2 and does the following operations:

$$Y_2 \| X_1 = K_2 \| K_1 \oplus Q \quad (4)$$

$$N_B = Y_2 \oplus X_1 \oplus K_1 \quad (5)$$

$$N_A = X_1 \oplus K_2 \quad (6)$$

5. Once the nonce N_A and N_B are extracted, U_i recalculates the hash message using the retrieved nonce and compares it with the received one. If the verification does not hold, U_i terminates the session. Otherwise, U_i generates a random nonce N_C, a substring serves as a new challenge C' (C' is obtained from N_C) and subsequently it computes $R' = PUF(C')$ using its PUF. These new generated parameters are encoded as follows:

$$M' = R' \oplus K_2 \| K_1 \quad (7)$$

$$N' = N_C \oplus K_1 \quad (8)$$

The session key with which the two entities will communication is computed as follows:

$$Sk_i = (K_1 \oplus N_B) \| (K_2 \oplus N_C) \quad (9)$$

6. Then, U_i sends the message $M_3 = \{M', N', H(R \| TUID_i \| N_B \| N_C \| Sk_i)\}$ to GS.
7. Upon the reception of M_3, GS obtains the new challenge-response pair and the session key:

$$N_C = N' \oplus K_1 \quad (10)$$

$$R' = M' \oplus (K_2 \| K_1) \quad (11)$$

$$Sk_i = (K_1 \oplus N_B) \| (K_2 \oplus N_C) \quad (12)$$

8. Afterward, GS checks the hash value $H(R' \| TUID_i \| N_B \| N_C \| Sk_i)$ using the retrieved parameters. If the verification is unsuccessful, GS terminates the session. Otherwise, GS memorizes the new challenge response pair (C', R') along with the old pair in its DB. At this stage, the mutual authentication is completed, so GS can start a secure transmission with U_i using the shared session key Sk_i.
9. In the other hand, both U_i and GS update the temporary identity $TUID_i$ for their subsequent authentication rounds as follows:

$$TUID_{i+1} = H(K_2 \| TUID_i \| K_1) \quad (13)$$

10. This phase ends with an acknowledgement string Ack along with a hash $H(Ack || GID || N_C)$ transferred from the GS to address the de-synchronization issue.

3.1.3 UAV-UAV Authentication

This phase describes how any two given UAVs could open a secure transmission session UAV-UAV while basing itself on the above described UAV-GS authentication scheme. The different steps of this phase are given as follows:

1. When a secure session is achieved between the UAV_{U_1} and GS using Sk_1, U_1 requests GS for a secure session with a second UAV_{U_2}. At this stage, GS sends an authentication request to the $UAV\ U_2$ that includes $\{Req, H(Req \| TUID_2 \| GID)\}$.
2. Afterward, U_2 checks the validity of $H(Req \| TUID_2 \| GID)$ and starts the same authentication process described in the above section, to establish a secure transmission session with GS using the shared session key Sk_2.
3. Subsequently, GS produces a new secret key Sk_{12} and transmits it encrypted to both U_1 and U_2 using the shared session keys Sk_1 and Sk_2, respectively. Finally, both UAVs share the same Sk_{12} and thus a secure communication channel is established between the two UAVs.

3.2 Session Key Disclosure Attack

In this section, we show that Alladi et al.'s scheme is vulnerable to eavesdropping attack and secret disclosure attack. According to the attack model assumed by the authors, an attacker ($\mathcal{A}$) can eavesdrop on the communication between the U_i and GS where he has access to the exchanged messages. This spying allows to the attacker to combine the broadcasted messages, such as the parameter Q and M' to reveal the session key Sk_i shared between the UAV and GS and that shared between the two UAV (Sk_{12}) during the UAV-UAV authentication phase.

3.2.1 Session Key Disclosure Attack (UAV-GS)

The session key Sk_i is supposed to be a secret parameter shared only between the *GS* and the UAV to ensure a secure communication session. In fact, the disclosure of this shared key will allow to the attacker to decrypt all the communications between the two entities and then get all the secret exchanged data during the communication. The different steps of this attack are given below:
As the authors of (Alladi et al., 2020a) have misused or badly combined the concatenation and the XOR operations in equation (14) from the M_2, this has led to the following simplification of this equation (as showed in figure (1)), using mathematical properties between XOR and concatenation operations, knowing that X_1, Y_2, K_1 and K_2 are 160 bit length.

$$Q = (Y_2 \| X_1) \oplus (K_2 \| K_1) \quad (14)$$

$$Q = (Y_2 \| X_1) \oplus (K_2 \| K_1) = (Y_2 \oplus K_2) \| (X_1 \oplus K_1) \quad (15)$$

On top of that, knowing that (from equations (1) and

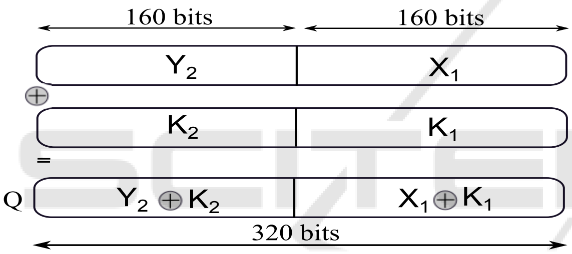

Figure 1: The simplification of the equation (14).

(2)): $X_1 = N_A \oplus K_2$ and $Y_2 = N_B \oplus X_1 \oplus K_1$
We can shorten the message Q as follows:

$$Q = (N_B \oplus (N_A \oplus K_2) \oplus K_1 \oplus K_2) \| (N_A \oplus K_2 \oplus K_1) \quad (16)$$

$$Q = (N_B \oplus N_A \oplus K_1) \| (N_A \oplus K_2 \oplus K_1)$$

It can be easily seen that Q can be splited into two parts q_1 and q_2 given as follows: $Q = q_1 \| q_2$ where:

$$q_1 = N_B \oplus N_A \oplus K_1 \quad (17)$$

$$q_2 = N_A \oplus K_2 \oplus K_1 \quad (18)$$

Accordingly, as the parameters N_A and Q are public, we can calculate the following quantities:

$$Q_1 = q_1 \oplus N_A = N_B \oplus K_1 \quad (19)$$

$$Q_2 = q_2 \oplus N_A = K_2 \oplus K_1 \quad (20)$$

The parameter Q_1 appears to be the first part of the session key Sk_i (knowing that $Sk_i = (K_1 \oplus N_B) \| (K_2 \oplus N_C)$). While the second part of Sk_i can be derived by combining equations (8) and (20) as follows: $N' \oplus (Q_2) = N_C \oplus K_1 \oplus (K_2 \oplus K_1) = N_C \oplus K_2$. Then, the session key is obtained.

Finally, using the disclosed Sk_i between the *UAV* U_i and the *GS*, the attacker could decrypt and exploit all the sensitive and critical information exchanged between the two entities which could lead to harmful impacts. Consequently, Alladi et al.'s scheme is vulnerable to session key disclosure attack which can be used to decrypt all the sensitive data exchanged via the insecure channel.

3.2.2 Session Key Disclosure Attack (UAV-UAV)

In this subsection, we show how an attacker can exploit the session key disclosure attack of Sk_i between the *UAV* U_i and the *GS*, described above, to disclose the secret session key Sk_{12} shared between the *UAV* U_1 and *UAV* U_2 and then decrypt the UAV-UAV communications. The steps of this attack are given in the following:
▷ When UAV_{U_1} requests *GS* for a secure session with a second UAV, *GS* transmits an authentication request to an appropriate *UAV* U_2.
▷ Thereafter, *UAV* U_2 starts the same authentication process (UAV_{U_2}-GS) to establish a secure session with *GS* i.e. produces the session key Sk_2.
▷ In this case, the attacker follows the same session disclosure attack, given above, to extract the secret key Sk_2.
▷ Subsequently, *GS* generates a new secret key Sk_{12} and transmits it encrypted to both U_1 and U_2 using Sk_1 and Sk_2, respectively.
▷ Therefore, the attacker decrypts the secret key Sk_{12} using the disclosed session keys Sk_1 or Sk_2.
Consequently, revealing the session key Sk_{12} allows to the attacker decrypting all the exchanged messages between the two UAVs, which makes the communication session insecure. This attack could have serious concerns on the mission course, especially for the case of military or strategic missions.

4 THE IMPROVED IoD SCHEME

In this section, we describe our suggested improved version of Alladi et al. scheme. In this enhanced scheme, we put forward efficient countermeasures to overcome the revealed flaws and then ensure a secure mutual authentication and key agreement between the different communicating entities. Consequently, the improved IoD authentication scheme includes three major phases as in the original scheme SecAuthUAV of Alladi et al. (Alladi et al., 2020a). In the improved version, we assume the same network and adversary models as in the original paper. Besides, we maintains the same assumptions.

4.1 Introduced Countermeasures

The critical weakness of Alladi et al. scheme is essentially related to the session key generating procedure that is not deeply examined and also to the misuse of the combination of XOR and concatenation operations employed to introduce diffusion on the different exchanged messages. These flaws largely facilitated the computation of the session key for the attacker. As a result, in the enhanced version we suggest a new expression for the session key computation while we keep almost all the steps of the three phases of SecAuthUAV scheme without change. Accordingly, we suggest the following new expression to compute Sk_i: In step (5) of the UAV-GS authentication phase, the user U_i and the GS have to proceed as follows:
In both sides, GS and UAV have to split the PUF output R' into K'_1 and K'_2 and compute the session key Sk_i according to the new equation:
$Sk_i = H(N_B \| N_C \| (K'_1 \oplus N_B \oplus K_1) \| (K'_2 \oplus K_2 \oplus N_C)$. In fact, with this new session key formula, it is difficult for an attacker to reveal any useful information from the exchanged public messages or from the new value of Sk_i which contains both ephemeral secrets and random variables secured by the hash function. Alongside, with the new equation of Sk_i, it is difficult for the attacker to construct it. On the other hand, even though the new scheme adds an extra computation by summing up a hash function, this does not affect the whole scheme. On the contrary, it provides additional security features and a high security level for the application that makes it hard to break. Additionally, we maintain the same steps in the UAV-UAV authentication phase as the revealed session key disclosure attack for this phase are caused essentially by the vulnerability of the UAV-GS authentication phase fixed above.

4.2 Security Analysis

The main objective of this section is to prove that the improved version provides all the security features claimed by Alladi et al.(Alladi et al., 2020a) and in addition, show its resistance against to the described attack. In this context, numerous security analysis tools via formal and informal models are used in the literature to check the robustness and security level of authentication schemes. These security tools can include Mao Boyd logic (Paulson, 1997), BAN (Abadi and Needham) logic (Agray et al., 2001), AVISPA model (Vigano, 2006), random and dynamic oracle models (Ene et al., 2009), etc. For this scheme, we perform this step using the well-accepted random oracle model as defined in (Canetti et al., 2004) by demonstrating that both the two authentication phases of the new scheme are secure against session key disclosure attack. So, we assume the following random oracle for the attacker $\mathcal{A}$:
Reveal. The *Reveal* random oracle will totaly output the string x from the corresponding hash value y, i.e. $y = H(x)$.

Proposition 1. Under the PUF function $P(\cdot)$ and the one-way hash function $H(\cdot)$ which acts as random oracle, our improved UAV-GS authentication is secure against an attacker $\hat{\mathcal{A}}$ disclosing the session key Sk_i and GS's identity GID.

Algorithm 1: $\text{Exp}1^{Hash,PUF}_{\hat{\mathcal{A}},Im-IoD}$.

1-Eavesdrop on the insecure channel and intercept ($M_1 = \{TUID_i, N_A, H(R\|TUID_i\|N_A)\}$, $M_2 = \{Q, H(Q\|GID\|N_A\|N_B)\}$ and $M_3 = \{M', N', H(R\|TUID_i\|N_B\|N_C\|Sk_i)\}$).
2-Call Reveal oracle on input $H(R\|TUID_i\|N_A)$. Let $(R') \leftarrow \texttt{Reveal 1}H(R\|TUID_i\|N_A)$
3-Compute K_1 and K_2 and then extract N_A and N_B.
4-Call Reveal oracle 1 on input $H(Q\|GID\|N_A\|N_B)$. Let $(GID') \leftarrow \texttt{Reveal 1}H(Q\|GID\|N_A\|N_B)$.
if $GID' = GID$ **then**
    Accept GID' as the GS's identity.
    5-Call Reveal oracle on input $H(R\|TUID_i\|N_B\|N_C\|Sk_i)$. Let $(Sk'_i) \leftarrow \texttt{Reveal 1}H(R\|TUID_i\|N_B\|N_C\|Sk_i)$
    if $Sk'_i = Sk_i$ **then**
        Accept Sk'_i as the session key Sk_i between UAV-GS.
    else
        Return 0 (Failure)
    end if
else
    Return 0 (Failure)
end if

Proof: Consider an attacker $\mathcal{A}$ with capabilities to disclose the shared session key Sk_i between the UAV and GS and get the GS's identity GID. For this, $\mathcal{A}$ initiates the algorithm experiment $Exp1^{Hash}_{\hat{\mathcal{A}},Im-IoD}$ given in Algorithm 1 against the improved IoD scheme, say Im-IoD by simulating the reveal Oracle 1. We express the success probability of the above given experiment as $succ_1 = |Pr[Exp1^{Hash}_{\hat{\mathcal{A}},\texttt{Im-IoD}} = 1] - 1|$. Besides, the advantage supported by $\mathcal{A}$ is expressed as $Adv1^{Hash}_{\hat{\mathcal{A}},\texttt{Im-IoD}}(t, q_{rev}) = \underset{\hat{\mathcal{A}}}{Max}\{succ_1\}$, where $\mathcal{A}$ can launch maximum *Reveal* queries q_{rev}. as stated in $Exp1^{Hash}_{\hat{\mathcal{A}},Im-IoD}$, $\mathcal{A}$ is able to divulge the shared session key Sk_i and the GS's identity GID only if he has the ability to invert the one-way hash function. Conversely, according to the *definition*, it is computationally untractable for $\mathcal{A}$ to break the one-way function and the win the game, i.e. $Adv_{\hat{\mathcal{A}}}^{Hash}(t) \leq \varepsilon$, for any sufficiently small $\varepsilon > 0$. Therefore, $Adv1_{\hat{\mathcal{A}},\texttt{Im-IoD}}^{Hash}(t, q_{rev}) \leq \varepsilon$. Consequently, our enhanced scheme is invincible against $\mathcal{A}$ disclosing the session key Sk_i between the UAV-GS and the GS's identity.

Proposition 2. Based the one-way hash function $H(\cdot)$ which acts as random oracle, our enhanced UAV-UAV authentication is secure against $\hat{\mathcal{A}}$ extracting the session key Sk_{12}.

Algorithm 2: $\text{Exp2}_{\hat{\mathcal{A}},Im-IoD}^{Hash}$.

1-Eavesdrop on the public channel between $UAV_1 - GS$ or $UAV_2 - GS$ and intercept the exchanged public messages (ex. for $UAV_1 - GS$: $M_1 = \{TUID_i, N_A, H(R\|TUID_i\|N_A)\}$, $M_2 = \{Q, H(Q\|GID\|N_A\|N_B)\}$, $M_3 = \{M', N', H(R\|TUID_i\|N_B\|N_C\|Sk_i)\}$),
$\{\texttt{Session key}\, Sk_{12}\}_{Sk_1}$ and $\{\texttt{Session key}\, Sk_{12}\}_{Sk_2}$.
2-Call Reveal oracle on input $H(R\|TUID_i\|N_B\|N_C\|Sk_i)$.
Let $(Sk_i') \leftarrow \texttt{Reveal 1}\, H(R\|TUID_i\|N_B\|N_C\|Sk_i)$
if $Sk_i' = Sk_i$ **then**
  Accept Sk_i' as the shared secret key Sk_i between the UAV-GS.
  3-Extract Sk_{12}' from $\{\texttt{Session key}\, Sk_{12}\}_{Sk_1}$.
  if $Sk_{12}' = Sk_{12}$ **then**
    Accept Sk_{12} as the shared secret key between UAV_1 and UAV_2.
    Return 1 (Success)
  else
    Return 0 (Failure)
  end if
else
  Return 0 (Failure)
end if

Proof. Let's consider an attacker $\mathcal{A}$ who have the capacity to disclose the shared session key Sk_{12} between two UAV_1 and UAV_2 throughout the UAV-UAV authentication phase. To do that, $\mathcal{A}$ performs the experiment $Exp2_{\hat{\mathcal{A}},Im-IoD}^{Hash}$ specified in Algorithm 2 against the enhanced scheme, by performing the reveal Oracle 1. We define the success probability of the above given experiment as $succ_2 = |Pr[Exp1_{\hat{\mathcal{A}},\texttt{Im-IoD}}^{Hash} = 1] - 1|$. Besides, the advantage supported by $\mathcal{A}$ is given as $Adv2_{\hat{\mathcal{A}},\texttt{Im-IoD}}^{Hash}(t, q_{rev2}) = \underset{\hat{\mathcal{A}}}{Max}\{succ_2\}$, where $\mathcal{A}$ can send maximum *Reveal* queries q_{rev2}. Based on $Exp2_{\hat{\mathcal{A}},Im-IoD}^{Hash}$, $\mathcal{A}$ is able to disclose the shared session key Sk_{12} if he has the capacity to invert the one-way hash function. Reciprocally, according to the *definition 1*, it is computationally difficult for $\mathcal{A}$ to break the one-way function, i.e. $Adv_{\hat{\mathcal{A}}}^{Hash}(t) \leq \varepsilon$, for any insignificant $\varepsilon > 0$. As a result, $Adv2_{\hat{\mathcal{A}},\texttt{I-scheme}}^{Hash}(t, q_{rev2}) \leq \varepsilon$. Finally, our enhanced scheme is secure against $\mathcal{A}$ who trying to disclose the shared session key Sk_{12} between the two UAVs.

4.3 Performance Analysis and Comparison

Our enhanced IoD scheme inherits all the strengths of Alladi et al.'s scheme and in addition, it considers new countermeasure against the revealed pitfall. Thus, extra computational cost was added to provide additional security features by using a supplementary hash function in the calculation of the shared session key Sk_i. Besides, the communication and the storage costs of the enhanced scheme are similar to those of the original one. Finally, with the introduce enhancement, the improved scheme could resist to the following security attacks: masquerade attack, man in the middle attack, replay attack, de-synchronization attack, cloning attack, etc. In addition, it provides the following security requirements: the provision for session key establishment, user anonymity, mutual authentication, forward secrecy, etc.

5 CONCLUSION

In this paper, we thoroughly examined the security of Alladi et al.'s IoD authentication scheme, revealing a fundamental flaw that is generated from a misuse of the mathematical combination of the concatenation and XOR operations. We showed that an eavesdropping attack can disclose the shared secret session key between the UAV and the *GS*, as well as the shared secret between the two UAVs, which might induce major risks to the mission's course, particularly in the case of strategic applications. Besides, we have suggested an upgraded version that fixes the discovered flaw. We may conclude from these kind of flaws that new authentication scheme's designs should be thoroughly evaluated from both informal and formal perspectives, using well-known concepts and guidelines. Furthermore, we may learn that misusing even a secure cipher with powerful cryptographic functions (PUF, hash function) can exceedingly compromise the security and privacy of the entire application. Lastly, we hope that this study will assist authentication designers in evaluating and improving the security and the durability of their IoD authentication solutions.

REFERENCES

Agray, N., Van Der Hoek, W., and De Vink, E. (2001). On ban logics for industrial security protocols. In *International Workshop of Central and Eastern Europe on Multi-Agent Systems*, pages 29–36. Springer.

Ali, Z., Chaudhry, S. A., Ramzan, M. S., and Al-Turjman, F. (2020). Securing smart city surveillance: A lightweight authentication mechanism for unmanned vehicles. *IEEE Access*, 8:43711–43724.

Alladi, T., Bansal, G., Chamola, V., Guizani, M., et al. (2020a). Secauthuav: A novel authentication scheme for uav-ground station and uav-uav communication. *IEEE Transactions on Vehicular Technology*, 69(12):15068–15077.

Alladi, T., Chamola, V., Kumar, N., et al. (2020b). Parth: A two-stage lightweight mutual authentication protocol for uav surveillance networks. *Computer Communications*, 160:81–90.

Alsamhi, S. H., Ma, O., Ansari, M. S., and Almalki, F. A. (2019). Survey on collaborative smart drones and internet of things for improving smartness of smart cities. *Ieee Access*, 7:128125–128152.

Canetti, R., Goldreich, O., and Halevi, S. (2004). The random oracle methodology, revisited. *Journal of the ACM (JACM)*, 51(4):557–594.

Choudhary, G., Sharma, V., Gupta, T., Kim, J., and You, I. (2018). Internet of drones (iod): threats, vulnerability, and security perspectives. *arXivpreprint arXiv:1808.00203*.

Ene, C., Laskhnech, Y., and Ngo, V. C. (2009). Formal indistinguishability extended to the random oracle model. In *European Symposium on Research in Computer Security*, pages 555–570. Springer.

Frikken, K. B., Blanton, M., and Atallah, M. J. (2009). Robust authentication using physically unclonable functions. In *International Conference on Information Security*, pages 262–277. Springer.

Gope, P., Millwood, O., and Saxena, N. (2021). A provably secure authentication scheme for rfid-enabled uav applications. *Computer Communications*, 166:19–25.

Gope, P. and Sikdar, B. (2020). An efficient privacy-preserving authenticated key agreement scheme for edge-assisted internet of drones. *IEEE Transactions on Vehicular Technology*, 69(11):13621–13630.

Hussain, S., Chaudhry, S. A., Alomari, O. A., Alsharif, M. H., Khan, M. K., and Kumar, N. (2021). Amassing the security: An ecc-based authentication scheme for internet of drones. *IEEE Systems Journal*.

Jan, S., Qayum, F., and Khan, H. (2021). Design and analysis of lightweight authentication protocol for securing iod. *IEEE Access*, 9:69287–69306.

Lin, C., He, D., Kumar, N., Choo, K.-K. R., Vinel, A., and Huang, X. (2018). Security and privacy for the internet of drones: Challenges and solutions. *IEEE Communications Magazine*, 56(1):64–69.

Lv, Z. (2019). The security of internet of drones. *Computer Communications*, 148:208–214.

Paulson, L. C. (1997). Proving properties of security protocols by induction. In *Proceedings 10th Computer Security Foundations Workshop*, pages 70–83. IEEE.

Rogaway, P. and Shrimpton, T. (2004). Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. In *International workshop on fast software encryption*, pages 371–388. Springer.

Shi, J., Lu, Y., and Zhang, J. (2019). Approximation attacks on strong pufs. *IEEE transactions on computer-aided design of integrated circuits and systems*, 39(10):2138–2151.

Srinivas, J., Das, A. K., Kumar, N., and Rodrigues, J. J. (2019). Tcalas: Temporal credential-based anonymous lightweight authentication scheme for internet of drones environment. *IEEE Transactions on Vehicular Technology*, 68(7):6903–6916.

Tian, Y., Yuan, J., and Song, H. (2019). Efficient privacy-preserving authentication framework for edge-assisted internet of drones. *Journal of Information Security and Applications*, 48:102354.

Vigano, L. (2006). Automated security protocol analysis with the avispa tool. *Electronic Notes in Theoretical Computer Science*, 155:61–86.

Wang, A., Tan, W., Wen, Y., and Lao, Y. (2021). Nopuf: A novel puf design framework toward modeling attack resistant pufs. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 68(6):2508–2521.

Wu, L., Hu, Y., Zhang, K., Li, W., Xu, X., and Chang, W. (2022). Flam-puf: A response feedback-based lightweight anti-machine learning-attack puf. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*.

Yahuza, M., Idris, M. Y. I., Wahab, A. W. A., Nandy, T., Ahmedy, I. B., and Ramli, R. (2021). An edge assisted secure lightweight authentication technique for safe communication on the internet of drones network. *IEEE Access*, 9:31420–31440.

Yu, W. and Wen, Y. (2019). Efficient hybrid side-channel/machine learning attack on xor pufs. *Electronics Letters*, 55(20):1080–1082.

Zhang, Y., He, D., Li, L., and Chen, B. (2020). A lightweight authentication and key agreement scheme for internet of drones. *Computer Communications*, 154:455–464.