electronics

MDPI

Article

Research on Identity Authentication Scheme for UAV Communication Network

Tao Xia *,†, Menglin Wang *, Jun He †, Shaofeng Lin, Yongqi Shi and Liyuan Guo

College of Information and Communication, National University of Defense Technology, Wuhan 430010, China; hejun17c@nudt.edu.cn (J.H.); linshaofeng17@nudt.edu.cn (S.L.); shiyongqi17@nudt.edu.cn (Y.S.); guoliyuan14@nudt.edu.cn (L.G.)

* Correspondence: xiatao17@nudt.edu.cn (T.X.); wangmenglin17@nudt.edu.cn (M.W.)

† These authors contributed equally to this work.

Abstract: Unmanned aerial vehicles (UAV) play a vital role in many fields, such as agricultural planting, security patrol, emergency rescue, and so on. The development and implementation of these devices have become vital in terms of reachability and usability. Unfortunately, as drones become more widely used in various fields, they become more and more vulnerable to attacks and security threats, including, but not limited to, eavesdropping, man-in-the-middle attacks, and known session key attacks. In order to deal with these attacks and security threats and meet the needs of lightweight UAV communication, a secure and efficient authentication scheme is essential. To meet the security and lightweight requirements of an identity authentication scheme in a UAV communication network, this paper proposes an identity authentication scheme sdronelig based on an elliptic curve cryptosystem. The scheme realizes the mutual authentication and session key agreement configuration between the UAV and the ground station, and the authentication and key agreement between the UAVs can be realized with the help of the control station. The sdronelig authentication scheme is based on the ECDH key exchange protocol in the elliptic curve cryptography algorithm and adopts the MAC message authentication code technology and the method of pre-calculating part of the process. Under the premise of ensuring the security of the UAV communication network, the authentication efficiency is improved, the communication overhead and communication times are reduced, and the lightweight requirement of the UAV authentication scheme is met. Additionally, a formal verification tool is used to verify the security of the sdronelig scheme under the Dolev-Yao threat model, which is suitable for UAV networks. Finally, a detailed comparative study was conducted on security features, communication overhead, the number of communications, and computational overhead. The results show that the proposed sdronelig authentication scheme not only provides superior security features but also has better or comparable overhead compared to other existing authentication schemes.

Keywords: UAV; identity authentication; security; formal verification; elliptic curve cryptography

Citation: Xia, T.; Wang, M.; He, J.; Lin, S.; Shi, Y.; Guo, L. Research on Identity Authentication Scheme for UAV Communication Network. *Electronics* **2023**, *12*, 2917. https://doi.org/10.3390/electronics12132917

Academic Editor: Stefano Scanzio

Received: 30 May 2023
Revised: 26 June 2023
Accepted: 29 June 2023
Published: 3 July 2023

1. Introduction

With the continuous progress of communication technology and the improvement of manufacturing capacity, UAVs have been widely used in military and civil fields. Today, drones play an important role in logistics distribution, agricultural planting, security patrol, emergency rescue, and other fields [1,2].

Due to the strong openness of the wireless data communication channel of the UAV network, the communication environment of the UAV system is more vulnerable to various attacks than other communication networks, including eavesdropping attacks, man-in-the-middle attacks, replay attacks, etc., [3]. At present, the security problems faced by UAV communication are seriously hindering the expansion of its application and the efficiency of its use. The identity authentication protocol between the UAV and the control station, as

the core technology to ensure communication security, can not only verify the legitimacy of the communication parties but also negotiate the session key for secure communication [4]. The DTLS protocol, the TLS protocol, and some existing UAV key protocols have some problems, such as excessive communication overhead, long authentication times, or insufficient security. The ECDH key exchange protocol is a method for two communication parties to establish a shared key over an insecure communication channel. It is based on Diffie-Hellman key exchange but uses elliptic curve cryptography (ECC) to generate and exchange keys. This protocol is commonly used in end-to-end encryption systems to ensure confidentiality and privacy. MAC authentication technology is a method to verify the integrity and authenticity of messages transmitted over the network. It involves adding a hash value or message authentication code (MAC) to a message using a key known to both communicating parties. The recipient can then verify this code to ensure that the message that has been transmitted has not been tampered with or forged. MAC authentication is commonly used in network protocols to ensure the authenticity and integrity of data packets. The combination of ECDH and MAC algorithms can replace digital certificates for identity authentication and reduce communication overhead and authentication time. The main research work of this paper is as follows:

- An identity authentication protocol called sdronelig has been proposed and designed for UAV and control station communication. The aim of the protocol is to achieve reduced communication times and overhead while ensuring security. Certificateless authentication techniques are employed, and part of the operation process is completed during the preprocessing stage, effectively reducing computational overhead and improving authentication efficiency;
- To ensure the security of the sdronelig scheme, a threat model suitable for UAV networks is designed based on the Dolev-Yao threat model [5]. The security of the sdronelig scheme is verified under this threat model using the formal verification tool ProVerif [6];
- Experiments were conducted to evaluate the performance of the sdronelig authentication scheme in two aspects. Firstly, the authentication time of the sdronelig authentication scheme was measured along with a comparison scheme in a simulated environment using Gazebo and Pycharm. Secondly, the communication overhead of the sdronelig authentication scheme was evaluated using Wireshark;
- A detailed comparative analysis of communication, computation overheads, and functionality attributes shows the superiority of the proposed scheme in terms of its provided security features, no special hardware required, and comparable or better communication and computational overheads as compared to those for other existing and relevant competing schemes in the literature.

The remainder of the paper is organized as follows: Section 2 presents an overview of related work. Section 3 describes the UAV communication network models and threat models. Section 4 elaborates on the implementation steps and calculation method of the authentication scheme. In Section 5, the security of the scheme is evaluated using Proverif. The performance of the scheme is assessed in Section 6. In Section 7, the paper concludes with a summary of the results.

2. Reference Review

In order to ensure the security of wireless communication in Unmanned Aerial Vehicles (UAVs), a significant amount of research has been conducted both domestically and abroad. The security protection schemes for UAV communication are primarily derived from the physical layer and cryptography.

Recent works [7–9] have designed UAV communication security protection schemes based on the physical layer that are suitable for addressing the characteristics of UAV nodes such as limited computing power and high dynamic changes in network topology. These schemes effectively utilize UAV mobility to enhance communication security. However, physical layer-based security schemes primarily serve as a supplement to traditional

cryptography-based security protection schemes [10]. Such schemes cannot guarantee the integrity of the transmitted information or the legitimacy of both communication parties' identities. Therefore, they cannot be employed in isolation to ensure the wireless communication of UAVs with high security requirements.

From the perspective of cryptography, the security protection scheme is primarily designed to complete the identity authentication of both communicating parties and negotiate a secure session key. This scheme involves combining various cryptographic algorithms based on mathematical problems to encrypt transmitted data. Currently, the widely used TLS and DTLS communication protocols on the Internet can ensure secure communication. However, they suffer from excessive communication overhead, computing overhead, and storage overhead, which do not meet the requirements for lightweight and efficient UAV communication. Security protection schemes based on cryptography primarily revolve around the completion of identity authentication for both communicating parties. These schemes negotiate a secure session key by combining various cryptographic algorithms, which are based on mathematical problems, to encrypt the transmitted data.

Azza Allouch et al. [11] improved the MAVLink protocol by incorporating encryption algorithms to ensure secure communication between UAVs and ground stations. However, this protocol only enables encrypted communication, lacking identity authentication and key agreement, which does not protect against man-in-the-middle attacks, impersonation attacks, and other attacks. As a mobile ad hoc network, UAVs possess three authentication protocol requirements: lightweight, high efficiency, and security. For identity authentication, UAVs can generate a unique identification mark based on their channels, circuits, and other distinguishing characteristics to identify their identity. For example, Kim et al. [12] have designed a security protection scheme for detecting UAVs using sound data, but such schemes require the use of a certificate authority, which tends to occupy significant storage space and computing resources. Improving upon this method, recent works [13,14] have proposed a lightweight authentication scheme based on PUF (Physical Unclonable Function), which reduces storage and computation overheads and meets the lightweight requirements. However, PUF technology on UAVs has some shortcomings, such as its immature technology, increased production costs of UAVs, and inconsistency with the hardware conditions of most UAVs.

Therefore, the cryptography-based design of UAV identity authentication schemes remains the mainstream development direction. Kwanwoong Yoon et al. [15] investigate a solution to prevent anonymous attackers from hijacking network channels or physical hardware on commercial drones. This scheme involves additional encrypted communication channels, authentication algorithms, and DoS attacks to maintain control of the drone during hijacking problems. However, it remains vulnerable to replay attacks. Srinivas et al. [16] propose a "lightweight authentication scheme based on time credentials in UAV environments". Nevertheless, Zeeshan Ali et al. [17] have demonstrated that this scheme does not prevent impersonation attacks and has security flaws. Haqi Khalid et al. [18] proposed a lightweight anonymous two-factor authentication scheme for UAVs based on asymmetric cryptographic methods, but this scheme lacks formal security analysis. Saeed Ullah Jan et al. [19] verified the security of the protocol through the ProVerif2.02 model and analyzed the storage, computation, and communication overhead to solve the performance problem of the security protocol, but Abdelouahid et al. [20] considered that it did not solve the problem of Known session key attacks. Zeeshan Ali et al. [17] proposed a "lightweight authentication mechanism for drones in smart city environments". However, B.D. Deebak et al. [21] have pointed out that Ali et al.'s scheme cannot withstand session key disclosure attacks. Furthermore, the security schemes in [22–24] are all susceptible to known session key attacks. To tackle these issues, the introduction of the ECDH protocol has become the mainstream choice. ECDH is a key exchange protocol with high security, and even with a known session key attack, it is difficult to compromise its security. Yongho Ko et al. [25] proposed a secure communication protocol for protecting UAVs and ground control stations based on the ECDH protocol. The scheme combines ECDH and the identity certificate issued by the

organization for key agreement to ensure the security of forward security and two-way authentication. Session key disclosure attacks are difficult to compromise, and denial of service attacks are partially mitigated. However, this scheme requires a certificate authority, which consumes a lot of resources, and it increases the communication overhead by passing certificates multiple times during the communication. In order to solve these problems, Jian Qirui et al. [26] canceled the use of identity certificates and designed an authentication and key protocol for UAVs and control stations based on the ECDH algorithm, which reduced the computational and communication overhead. However, the authentication scheme designed in this scheme takes the control station as the initiator of identity authentication, whereas technically, the UAV should be the initiator [27–29]. In addition, the scheme is computationally cumbersome, has too many communications, and lacks an authentication method between drones. Therefore, by recombining ECDH and MAC algorithms, the authentication efficiency can be further improved and the computational overhead can be reduced. The identity authentication scheme sdronelig combines the ECDH and MAC algorithms to realize identity authentication between UAV and control station. The UAV is the initiator of the authentication process, which reduces communication times. On the basis of canceling the use of identity certificates, part of the ECDH calculation process is completed in advance in the preprocessing stage, which improves authentication efficiency.

3. UAV Communication Network Model and Threat Model

3.1. UAV Communication Network Model

A UAV network is a type of wireless network designed to complete tasks with UAV applications at its core. The network can be self-organized or supported by communication facilities such as ground control stations and satellites.

The identity authentication scheme presented in this paper is mainly targeted at the UAV communication network model depicted in Figure 1.

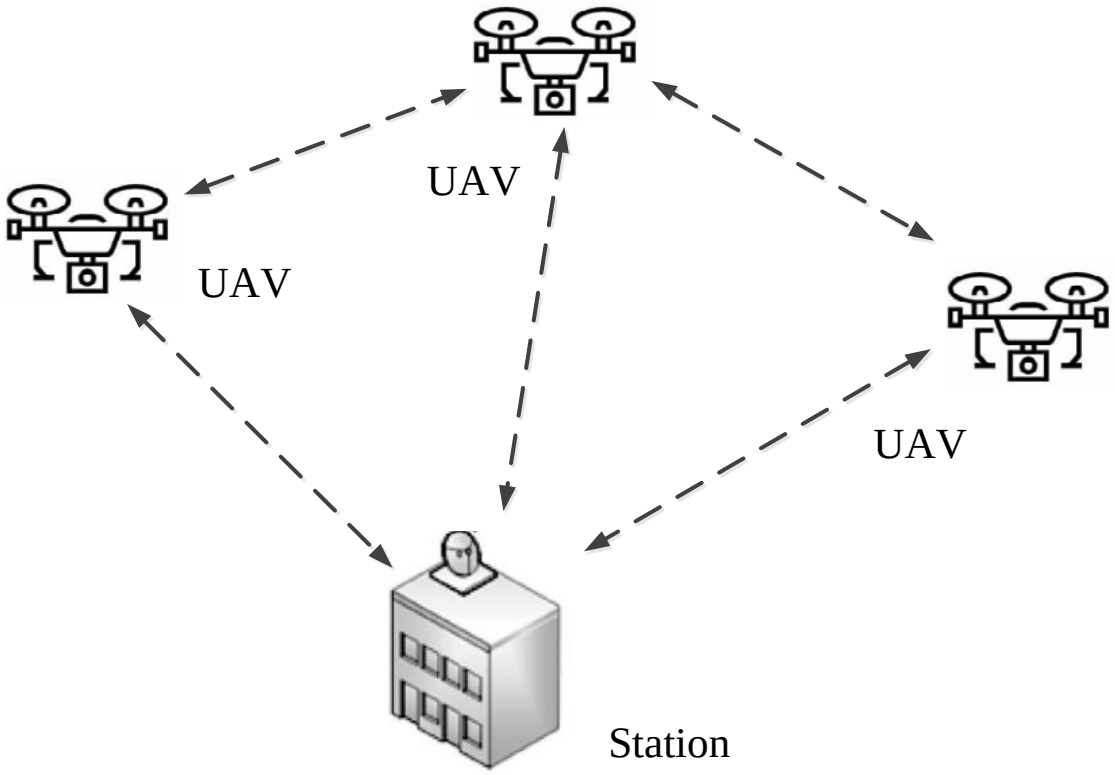

Figure 1. UAV communication network model diagram.

In this network, the ground control station is connected to multiple UAVs through wireless links to control and monitor multiple UAVs. The legal UAV may implement secure

encrypted communication with the control station through an identity authentication scheme. In addition, the legal UAV may implement encrypted communication with other legal UAVs through the control station after establishing a secure communication channel with the control station.

3.2. Threat Model

Compared to other wireless sensor networks, UAV networks have the characteristics of fast movement speeds and limited available energy. This makes their fragile communication channels more susceptible to security risks. Dolev and Yao [5] proposed a classical wireless network threat model that assumes that the attacker is very powerful, which is very suitable for designing this kind of wireless network threat model with low security. At present, the Dolev–Yao model has been widely used and has become the established standard of wireless network security protocol threat modeling. In this threat model, the attacker is extremely powerful, analogous to drones in the network communicating with the attacker; the messages received are also from the attacker. The Dolev–Yao model accurately describes the behavior of attackers, and the specific assumptions are shown in Table 1.

Table 1. Specific assumptions of the Dolev–Yao model.

Attacker Capabilities in the Dolev–Yao Model	Abilities Not Available to Attackers in Dolev–Yao Model
Information that traverse that communication network can be obtained	They do not have the ability to guess random numbers in a large enough space.
Have a legal identity in the communication network, and can impersonate other principals to initiate communication with any principal	In the absence of the correct key (or private key), the attacker does not have the ability to recover the plaintext from the given ciphertext; For the complete encryption algorithm, the attacker also does not have the ability to obtain the correct ciphertext from the given plaintext.
Become the recipient of any subject's message	The private part, e.g., the private key that matches the given public key, cannot be solved for.
impersonate any principal to send information to any other principal	While public portions of the communication environment may be controlled, private areas in the computing environment, such as the memory of an offline principal, are generally not controlled.

According to the special environment of the UAV network, in the preprocessing stage, that is, the stage before the UAV leaves the factory and the UAV takes off, the data exchange process between the UAV and the control station adopts a safe method. In practice, an attacker usually cannot enter this stage to obtain information, i.e., a legitimate identity, at this stage. This situation is contrary to the Dolev–Yao model assumption that the attacker has a legitimate identity in the communication network. Therefore, it is necessary to improve the Dolev–Yao model according to the actual situation and remove the strong assumption that the attacker can obtain a legal identity in the communication network. The situation that the attacker cannot obtain identity information in the initialization phase is also consistent with the assumption that the attacker cannot control the private area in the computing environment in the Dolev–Yao model. In addition, in order to better fit the characteristics of the UAV network, this paper assumes that the attacker can obtain the private keys of some legitimate nodes through other means, but not the legitimate ground station nodes. Specifically, the specific assumptions of the threat model in the UAV network are shown in Table 2.

Table 2. Specific assumptions of threat models in UAV networks.

Attacker Capabilities in UAV Communication Networks	Capabilities That an Attacker Does Not Have in a UAV Communications Network
Attackers can receive data transmitted over wireless channels between any node in the network	Does not have the ability to guess random numbers in a large enough space
Attackers can send data to any node in the network through wireless channels	Without the correct key, the attacker cannot recover the plaintext from the given ciphertext and cannot obtain the correct ciphertext from the given plaintext.
An attacker has the ability to derive a public key from a given private key	The attacker cannot derive the private key from the public key without unknown the parameters, and cannot generate a valid message authentication code for the message
The attacker can control a small number of nodes and obtain the private key of the controlled node by other means.	The attacker does not have the ability to hijack the ground control station and cannot obtain the key stored offline by the ground control station.

4. Certification Scheme

Based on the above communication network model and threat model, this paper designs the sdronelig authentication scheme to achieve the following security goals:

The legal ground station and the unmanned aerial vehicle node that successfully run the authentication scheme can realize two-way identity authentication between the legal ground station and the unmanned aerial vehicle node and effectively prevent any external or internal attacker from masquerading as a legal node to pass the authentication.

The legitimate ground station and the UAV node that successfully run the authentication scheme can negotiate a consistent key, and the key can only be obtained by the legitimate node and the attacker cannot obtain it.

The symbols and meanings used in the certification scheme are shown in Table 3.

Table 3. Symbols and meanings involved in the scheme.

Symbol	Implication
S	ground control station
U	Legitimate drones that require authentication
$A\|\|B$	Connect two data points A and B into one data point
G	base point of elliptic curve
d_S	Ground control station private key
P_S	Ground control station public key
d_U	UAV private key
P_U	UAV public key
ID_U	UAV identification
ID_S	Ground control station identification
r_S	Temporary private key generated by ground control station based on elliptic curve
R_S	Temporary public key generated by ground control station based on elliptic curve
r_U	Temporary private key generated by UAV based on elliptic curve
R_U	Temporary public key generated by UAV based on elliptic curve
$rand()$	random number generating function
$enc(x, y)$	Encrypt data y using x as the key
$dec(x, y)$	Decrypt data y using x as the key
$hmac(x, y)$	Compute the hash authentication code for data y using x as the key

4.1. Initialization

System initialization parameters before UAV and control station equipment certification include:

(1) Elliptic curve parameters for the UAV and control station, i.e., selecting an appropriate and safe elliptic curve for the UAV and control station.

(2) The parameters that the UAV shall store before certification include $\{ID_U, d_U, P_U, ID_S, P_S, R_U, r_U, vk, mac_{U1}\}$. where ID_U is the UAV identity, d_U is the

UAV private key, and P_U is the public key generated by the UAV according to Equation (1).

$$P_U = d_U * G \tag{1}$$

ID_S is the control station identity, P_S is the control station public key, and r_U is the random number generated by the UAV, and R_U is calculated according to Equation (2).

$$R_U = r_U * G \tag{2}$$

In order to calculate the vk (Key used for identity authentication during authentication between the unmanned aerial vehicle and the control station) required for identity authentication, it is generated by the UAV according to Equation (3).

$$vk = R_U + d_U * P_S \tag{3}$$

The message authentication code mac_{U1} is generated by the drone according to Equation (4).

$$mac_{U1} = hmac(vk, ID_U \| R_U) \tag{4}$$

Among the above parameters, only ID_S, P_S, ID_U and d_U are entered at the delivery stage, and other parameters are generated by the UAV subsequently.

(3) The parameters to be stored by the ground station mainly include two parts: one is its own identification and public and private keys $\{ID_S, d_S, P_S\}$, and the other is the relevant parameters of the unmanned aerial vehicle under the jurisdiction of the control station. The unmanned aerial vehicle U is taken as an example for the description herein, which mainly includes $\{ID_U, P_U, R_S, r_S\}$.

d_S is the control station private key, and P_S is the control station public key generated by the control station according to Equation (5).

$$P_S = d_S * G \tag{5}$$

The ground control station uses the generated plurality of random numbers as a temporary private key and calculates a temporary public key. Here, the temporary private key r_S corresponding to the unmanned aerial vehicle U is taken as an example and R_S is calculated and generated according to Equation (6).

$$R_S = r_S * G \tag{6}$$

Among the above parameters, only ID_S, P_U, ID_U and d_S are entered at the delivery stage, and other parameters are generated by the control station subsequently.

4.2. Protocol Execution Process

The protocol flow is initiated by the UAV, and the flow is shown in Figure 2.

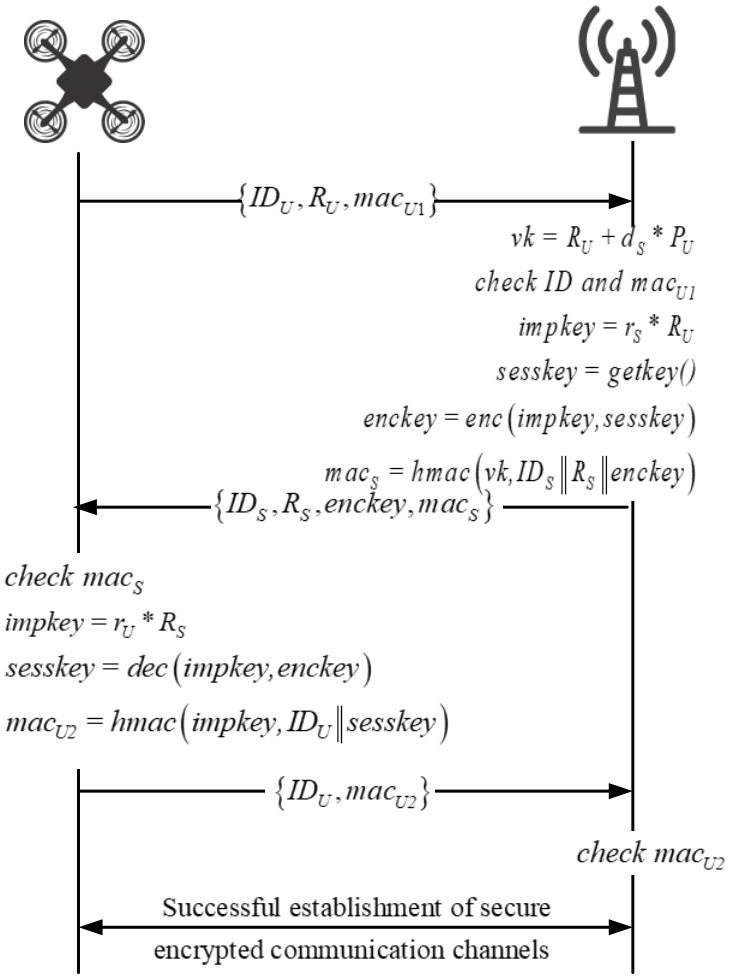

Figure 2. Authentication scheme flowchart.

Stage 1: The UAV initiates an authentication request to the control station.

Step 1-1: The UAV will send $\{ID_U, R_U, mac_{U1}\}$ to the control station.

Stage 2 After receiving and verifying the data, the control station sends the data to the UAV.

Step 2-1: The control station obtains the public key of the corresponding UAV according to the ID_U, and calculates vk. The calculation method of vk follows Equation (7).

$$vk = R_U + d_S * P_U \tag{7}$$

Step 2-2: The control station verifies mac_{U1}. The control station uses the received ID_U, R_U and the vk calculated in Step 2-1 to calculate the mac_{ver1} according to Equation (8). If the mac_{ver1} is the same as the mac_{U1}, the verification passes; otherwise, the authentication fails.

$$mac_{ver1} = hmac(vk, ID_U \| R_U) \tag{8}$$

Step 2-3: The control station uses the function $getkey()$ to randomly generate a key $sesskey$ that can be used for the symmetric encryption algorithm.

Step 2-4: $impkey$ ($impkey$ is a parameter used to calculate $enckey$) is calculated in accordance with Equation (9).

$$impkey = r_S * R_U \tag{9}$$

Step 2-5: Control station uses $impkey$ as key to encrypt $sesskey$ to generate $enckey$ according to algorithm. Calculation method of $enckey$ follows Formula (10).

$$enckey = enc(impkey, sesskey) \tag{10}$$

Step 2-6: The control station generates a message authentication code mac_S, which is calculated in accordance with Equation (11).

$$mac_S = hmac(vk, ID_S \| R_S \| enckey) \tag{11}$$

Step 2-7: After completing the above steps, the UAV sends the data $\{ID_S, R_S, enckey, mac_S\}$ to the control station.

Stage 3: After receiving and verifying the data, the UAV sends the data to the control station to confirm that the keys are the same.

Step 3-1: UAV verification mac_S. The UAV uses vk and the received $ID_S, R_S, enckey$ to calculate the mac_{ver2} according to Formula (12). If the mac_{ver2} is the same as the mac_S, the verification passes; otherwise, the verification fails.

$$mac_{ver2} = hmac(vk, ID_S \| R_S \| enckey) \tag{12}$$

Step 3-2: The UAV calculates $impkey$ according to Equation (13).

$$impkey = r_U * R_S \tag{13}$$

Step 3-3: The message $enckey$ is decrypted using the $impkey$ as the key to obtain the $sesskey$, and the calculation of the $sesskey$ follows Equation (14).

$$sesskey = dec(impkey, enckey) \tag{14}$$

Step 3-4: The drone is generated according to the formula, and the calculation method of mac_{U2} follows Formula (15).

$$mac_{U2} = hmac(tmpkey, ID_{\mathrm{U}} \| sesskey) \tag{15}$$

Step 3-5: The drone sends $\{ID_U, mac_{U2}\}$ to the control station.

Stage 4 After the control station receives the data and completes the data verification, the session key is used to establish a secure communication channel.

Step 4-1: The control station uses the received ID_{U}, $impkey$ generated in Step 2-4 and $sesskey$ generated in Step 2-3 to generate mac_{ver3} according to Equation (16). If mac_{ver3} is the same as mac_{U2}, authentication is successful.

$$mac_{ver3} = hmac(tmpkey, ID_{\mathrm{U}} \| sesskey) \tag{16}$$

Stage 5: Identity the authentication between drones.

After the UAV completes the identity authentication with the control station and establishes a secure communication channel, the UAV may realize the identity authentication with other UAVs by means of the control station, and the process is as shown in Figure 3.

After the UAV completes the identity authentication with the control station, the UAV can realize the identity authentication with other UAVs by means of the control station. Take UAV U_A and UAV U_B as an example to draw a flow chart, as shown in Figure 3.

The drone U_A generates a random number r_A, computes $R_A = r_A * G$ as a temporary public key, and forwards $\{ID_A, R_A\}$ to the drone U_B via the control station S over a secure communication channel.

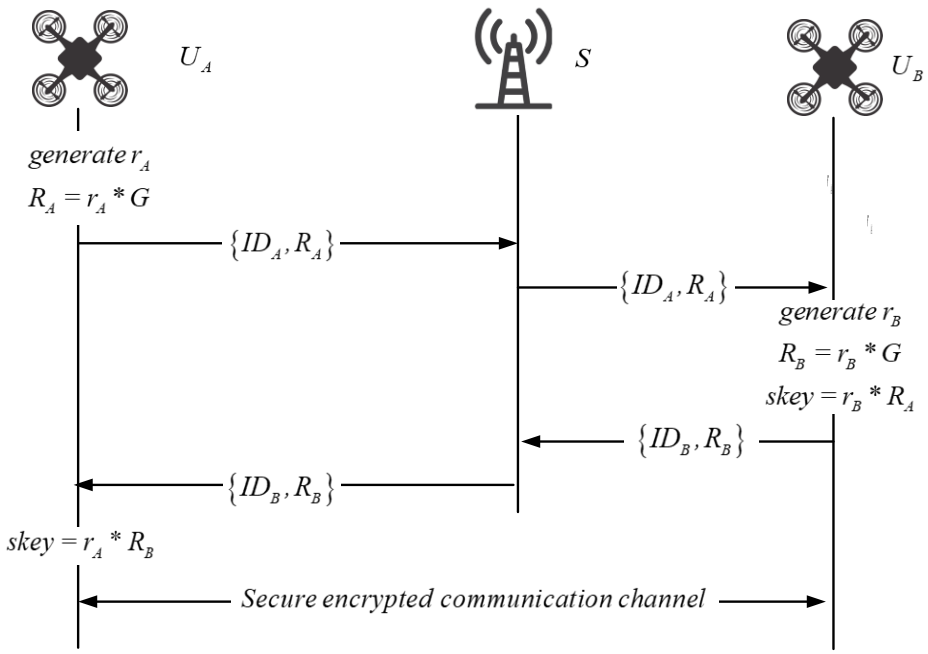

Figure 3. Flowchart of the authentication scheme between drones.

After receiving the authentication request from UAV U_A, UAV U_B first generates a random number r_B, then calculates $R_B = r_B * G$ as a temporary public key, then calculates a session key $skey = r_B * R_A$, and forwards $\{ID_B, R_B\}$ to UAV U_A via a secure communication channel of control station S.

After receiving the $\{ID_B, R_B\}$, the UAV U_A calculates a session key $skey = r_A * R_B$, and implements encrypted communication with the UAV U_B based on the ID_B and the $skey$.

5. Safety Analysis

In order to ensure the security of the protocol, this paper uses the formal analysis tool Proverif to analyze the security of the protocol. Its code is given in Appendix A of the paper. Proverif has been widely used in security proof of identity authentication schemes due to its advantages of automated verification, multilingual support, and high efficiency [19,27].

ProVerif is an event-based formal verification tool used to verify the correctness and security of protocols. When using this tool, you first need to define the key nodes of the protocol and then use a special typed process calculus language to describe the protocol, including processes, events, interactions, and constraints. By analyzing all possible event sequences in the protocol and checking whether these event sequences satisfy specific security properties, such as identity authentication, confidentiality, etc., according to formal specifications. Therefore, it is first necessary to define the key nodes of the protocol. Because the identity authentication phase between UAVs is carried out in the secure encrypted communication channel established between UAVs and the control station, this paper mainly proves the security of the identity authentication phase between UAVs and the control station. As shown in Table 4, the event accept_g_station indicates that the identity of the UAV has been authenticated at the ground station, and the ground station is about to send a final reply message to the UAV; the event accept_UAV indicates that the identity of the ground station has been authenticated at the UAV, and the UAV is about to send a final reply message to the control station; the event termi_g_station indicates that the ground station has finished executing the identity authentication protocol; and the event termi_UAV indicates that the authentication protocol has been executed by the UAV.

Table 4. Key node event.

Key Node Event
(* Parameter description: Key, whether hijacked *) event accept_g_station(key, bool). event accept_UAV(key, bool). event termi_g_station(key, bool). event termi_UAV(key, bool).

According to the above definition of key node events, the safety objectives given in Section 4 are defined using a special Typed Process Calculus language, as shown in Table 5. The first query statement is to verify the security of the key negotiated between the UAV and the control station; the second and third query statements are to verify whether the legitimate UAV and the control station node can normally execute the security scheme; and the fourth query statement is to verify that each time the UAV runs the security scheme, it completes authentication with the legitimate ground station ("inj-event" indicates that the two events described later are one-to-one, which verifies whether the security scheme is resistant to replay attacks).

Table 5. Safety objective description.

Safety Objective Description
(* Attacker cannot obtain secret key k *) query attacker(secrecy). (* Uncontrolled drones can execute the protocol normally *) query sk: key; event(termi_g_station(sk, false)). query sk: key; event(termi_UAV(sk, false)). (* Uncontrolled drones can complete mutual authentication and negotiate session keys *) query sk: key; inj-event(termi_g_station(sk, false)) ==> inj-event(accept_UAV(sk, false)). query sk: key; inj-event(termi_UAV(sk, false)) ==> inj-event(accept_g_station(sk, false)).

In order to verify that the UAVs controlled by the attacker cannot join the network, as shown in Table 6, before the authentication starts, the private keys of some UAVs (assuming that these UAVs have been attacked and controlled) are publicly output to the public channel, and the entities in the protocol description are extended to three parallel entities: Ground stations(g_station(id_s, gs_skey)), drones that are not attacked(UAV(id_d1,safe_skey, P_s, false)), and drones that are controlled by attackers(UAV(id_d2, compromised_skey, P_s,true)).

Table 6. Validation master process.

Validation Master Process

process
new id_s: bitstring;
new id_d1: bitstring;
new id_d2: bitstring;
let P_s = pk(gs_skey) in
let P_d1 = pk(safe_skey) in
out(c, P_s);
out(c, P_d1);
(* Publicize the private key of the controlled drone *)
out(c, compromised_skey);
(* Both legitimate drones and drones controlled by attackers can participate in the agreement *)
((!choose_UAV) |
(!g_station(id_s, gs_skey)) |
(!UAV(id_d1,safe_skey, P_s, false)) |
(!UAV(id_d2, compromised_skey, P_s,true)))

The safety verification results are shown in Table 7. The first "RESULT" indicates that the negotiated shared key is secure, that is, the attacker cannot obtain the shared key negotiated by both parties; the second and third "RESULT" indicate that the termi_g_station and termi_UAV security schemes can be executed normally and have the ability of identity authentication and key agreement, that is, they are alive; article 4 "RESULT" indicates that the legal UAV node can complete identity authentication at the ground station, that is, the UAV can confirm that the other party is a legal ground station when completing the protocol; the fifth "RESULT" indicates that the legal ground station node can confirm that the other party is a legal UAV when the UAV completes the identity authentication, that is, the ground station completes the protocol.

Table 7. Safety verification results.

Safety Verification Results
RESULT not attacker(secrecy[]) is true.
RESULT not event(termi_g_station(sk_3,false)) is false.
RESULT not event(termi_UAV(sk_3,false)) is false.
RESULT inj-event(termi_g_station(sk_3,false)) ==> inj-event(accept_UAV(sk_3,false)) is true.
RESULT inj-event(termi_UAV(sk_3,false)) ==> inj-event(accept_g_station(sk_3,false)) is true.

6. Experimental Simulation and Performance Analysis

6.1. Simulation Experiment

The experimental environment is divided into two parts: the control station and the UAV. The control station side is configured with an Intel(R) Core(TM) i7-7700HQ CPU@2.80 GHz and 2.81 GHz with 12GB of memory. Drone side is configured with an Intel(R) Core(TM) i7-7700HQ CPU@2.80GHz 2.81 GHz and 4GB RAM. Among them, the UAV terminal includes the quadrotor UAV simulation model built on the GazeBoo platform and the communication module built on the PyCharm 2021.1.1 x64 platform. The control station includes a communication module built on the PyCharm 2021.1.1 x64 platform. The main algorithms are written in Python, and the core algorithms at the control station end and UAV end are shown in Algorithms 1 and 2, respectively.

Algorithm 1: Control Station Core Algorithm

Output: If all return True output cost_time, False otherwise.
1. start_time = time.time()//the start point of that authentication process
2. S←U// S receives msg_u1 sent by U
3. id_u, R_u, mac_u1 <- Parsing msg_u1
4. vk<-R_u + self.sk * P_u
5. Compute HMAC = hmac.new(vk, id_u + R_u)
6. If HMAC == mac_u1 and station_id == id_u, return True.
7. Else, return False.
8. impkey <- r_s * R_u
9. key <- Use PBKDF2 algorithm to generate the user's key
10. ekey <- Encrypt the key using the AES algorithm
11. mac_s1<- hmac.new(vk, id + R_s+ekey).
12. msg_s1<- Packing [id, R_s, ekey, mac_s1]
13. S→U// S sends information msg_s1 to U
14. S←U// S receives msg_u2 sent by U

Algorithm 1: *Count*.

15. id_u2, key, mac_u2 <- Parsing msg_u2
16. Compute HMAC = hmac.new(tmpkey, id_u2 + key)
17. If HMAC == mac_u2 and station_id = id_u2, return True.
18. Else, return False.
19. end_time = time.time()//Authentication Process Endpoint
20. logger.info("Protocol finished in [%.2f]ms"%((end_time - start_time) * 1000))

Algorithm 2: UAV core algorithm

Output: If all return True output cost_time, False otherwise.
1. start_time=time.time()//the star point of that authentication process
2. msg_u1 = Packaging information [id, R_u, mac_u1])
3. U→S// U sends information msg_u1 to S
4. U←S// U receives msg_s1 sent by S
5. id_s, R_s, ekey,mac_u<- Parsing msg_s1
6. Compute HMAC = hmac.new(vk, id_s + R_s+ekey)
7. If HMAC == mac_s1 and station_id = id_s, return True.
8. Else, return False.
9. tmpkey <- r_u * R_s
10. key <-AES.new(tmpkey, AES.MODE_ECB).decrypt(ekey)
11. mac_g2 <- hmac.new(tmpkey, id + key + ktype)
12. msg_g2 <- Package [id, key, mac_g2]
13. U→S// U sends information msg_u2 to S
14. end_time = time.time()//Authentication Process Endpoint
15. logger.info("Protocol finished in [%.2f]ms"%((end_time - start_time) * 1000))
16. // Send takeoff command to drone model */
17. Pycharm→Gazeboo : command('fly')

In order to give consideration to security and operation speed, 256-bit prime field elliptic curve secp256r1 is used as an asymmetric encryption algorithm, 128-bit AES is used as a symmetric encryption algorithm, and HMAC based on SHA256 is used as a message authentication code algorithm. The UDP protocol is based on the Python socket interface to send and receive protocol data packets, which is convenient to provide fast transmission speed and less network load.

The SP-D2GCS protocol [25] is used in the comparative experiment, and the authentication process is shown in Figure 4.

Run the relevant program so that the UAV model takes off after receiving the command, as shown in Figure 5.

Run the authentication scheme for ten times, respectively, and record the data and draw them into histogram 6 and histogram 7, respectively. Figure 6 shows the time required for completing key configuration at the UAV end for two protocols under the same hardware environment (time from protocol execution to protocol completion), and Figure 7 shows the time required for completing key configuration at the control station end for two protocols under the same hardware environment (time from protocol execution to protocol completion). Calculating the average value, respectively, the time required for identity authentication on the UAV side is about 7.15 ms, which is about 88.93% less than the 64.63 ms of the SP-D2GCS [25] and more than 50% less than the 25.14 ms [26] of the DTLS protocol (a performance benchmark for comparing the efficiency of UAV security protocols [26]). The time required for identity authentication at the control station is about 6.95 ms, which is about 81.24% less than the 37.05 ms of SP-D2GCS [25] and more than 50% lower than the 25.83 ms [26] of the DTLS protocol. Therefore, the authentication scheme designed in this paper effectively improves the authentication efficiency compared with the DTLS protocol and the scheme of SP-D2GCS [25], which is also based on the ECC algorithm.

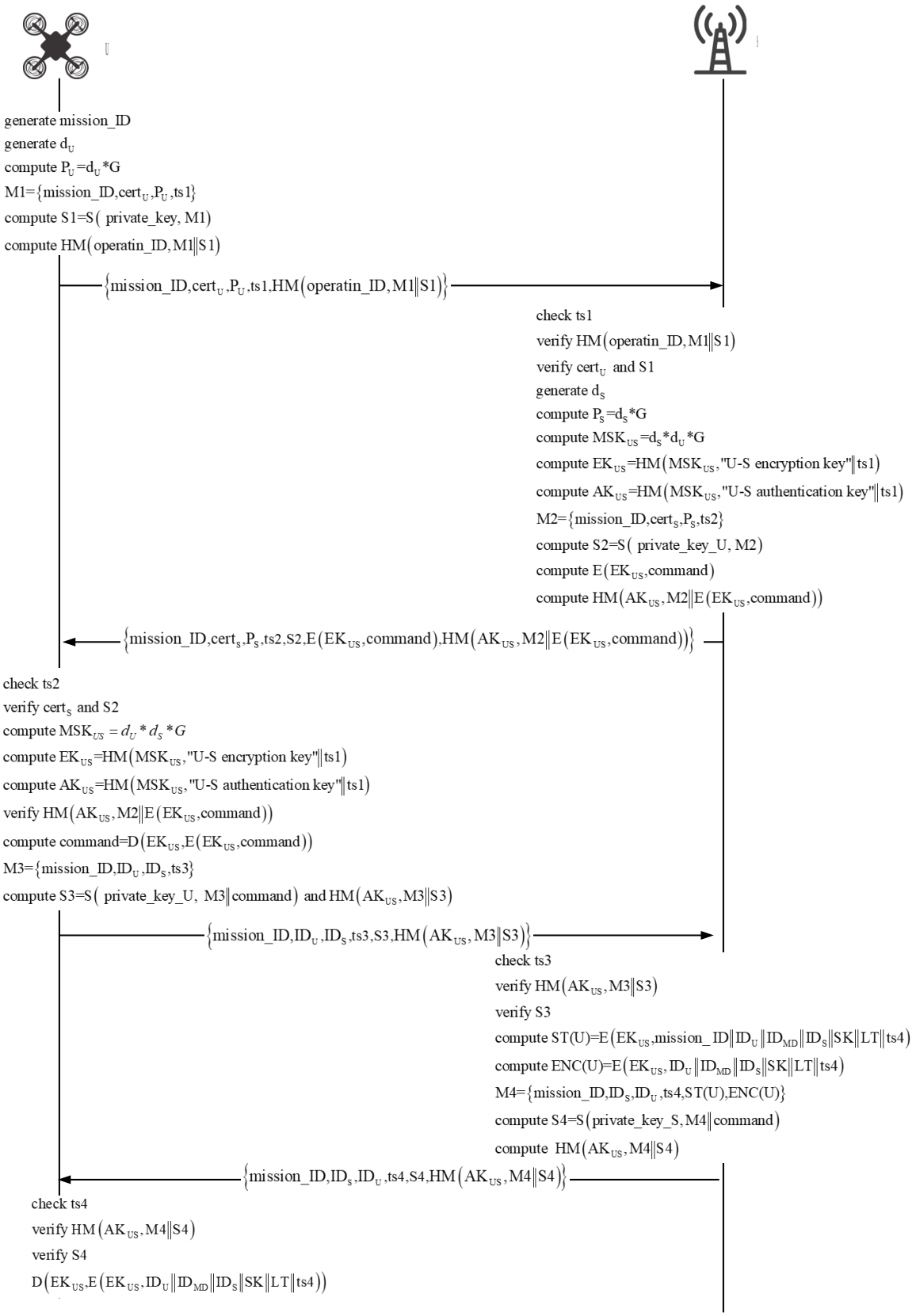

Figure 4. SP-D2GCS [25] certification flowchart.

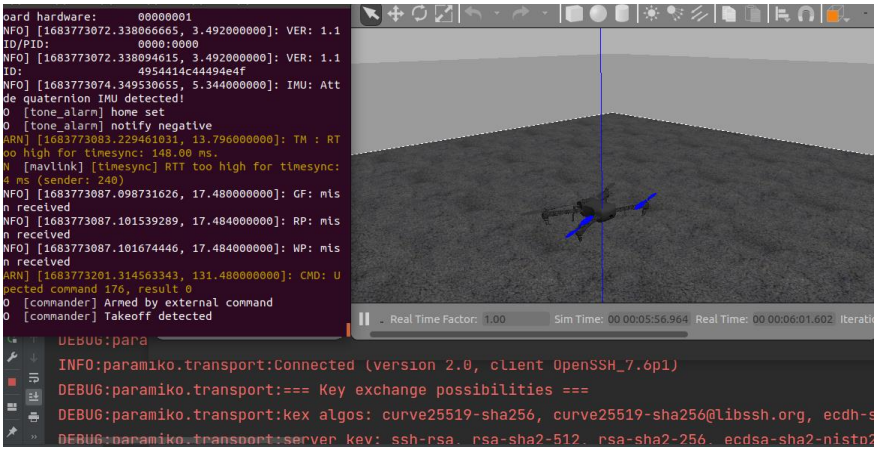

Figure 5. UAV model take-off screenshot.

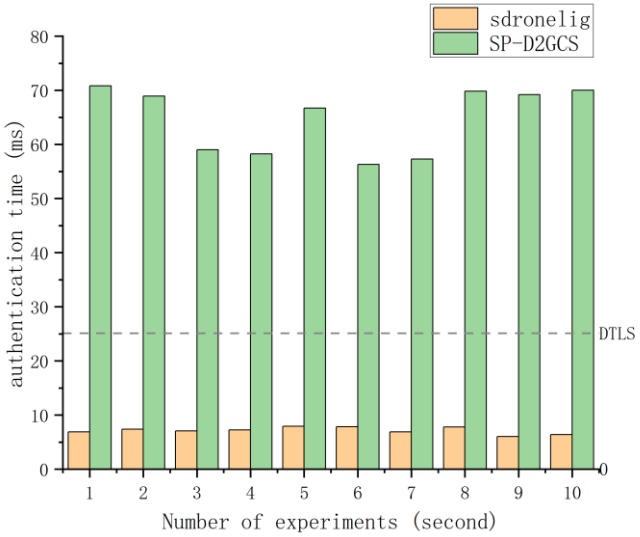

Figure 6. Comparison histogram of the UAV authentication time.

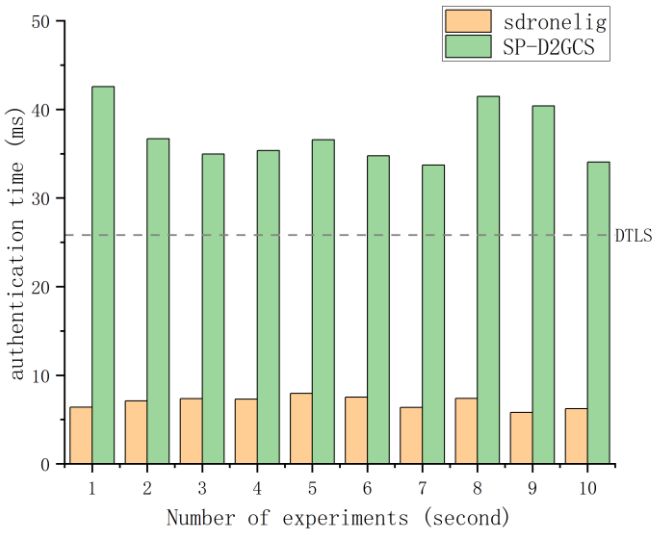

Figure 7. Comparison histogram of the station authentication time.

Use WireShark to capture the total amount of data sent during protocol interaction, and the result is shown in Figure 8. The calculated communication overhead is 301 bytes.The

sdronelig protocol was compared with Yongho et al. [25] and the DTLS protocol, and the results are shown in Figure 8.

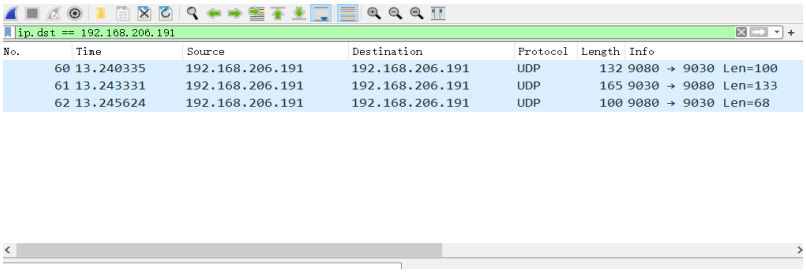

Figure 8. WireShark capture data interaction volume graph.

As shown in Table 8, after comparison with the original scheme, it is found that the communication overhead of this scheme is only 7.69% of the DTLS protocol and 12.48% of SP-D2GCS [25].

Table 8. Comparison table of communication expenses.

Identity Authentication Scheme	Communication Overhead (Bytes)
sdronelig	301
DTLS	3913
SP-D2GCS [25]	2411

6.2. Performance Analysis

This section compares the performance of sdronelig with that of several UAV authentication schemes proposed in recent years (since the authentication between UAV and control station is the basis of the whole network and some literature lacks the authentication between UAV and UAV, the authentication process between UAV and control station is mainly compared). The security characteristics of the selected representative UAV authentication scheme and the sdronelig scheme are compared, and the results are shown in Table 9.

Table 9. Comparison table of scheme security features.

Certification Scheme	Mutual Authentication	Effective against Replay Attacks	Effective against Man-in-the-Middle Attacks	No Special Hardware Required	Effective Response to Known Session Key Attacks
sdronelig	Yes	Yes	Yes	Yes	Yes
Chuang et al. [14]	Yes	Yes	Yes	No	Yes
Chin-Ling et al. [24]	Yes	-	-	Yes	No
SP-D2GCS [25]	Yes	Yes	Yes	Yes	Yes
DroneSec [26]	Yes	Yes	Yes	Yes	Yes
SENTINEL [29]	Yes	Yes	Yes	Yes	Yes
ACPBS-IoT [30]	Yes	Yes	Yes	Yes	Yes

Yongho [25], Jian [26], Cho [29], Basudeb [30], Chuang [14], et al. all explain the ability of their schemes to achieve mutual authentication, resist replay attacks, resist man-in-the-middle attacks, and resist known session key attacks. However, because the scheme designed by Chuang et al. [14] adopts PUF technology, it has special hardware requirements, including the need for programmable chips, high-demand circuit design, etc., which makes it difficult for ordinary UAVs to support this technology. The scheme designed by Chin-Ling et al. [24] does not adopt the ECDH algorithm and faces the threat of a known session key attack (Abdelouahid et al. [20] also think that it has defects when facing a known session key attack). It can be seen from Table 9 that the authentication scheme represented by Chuang et al. [14] has the defect of requiring special hardware, and Chin-Ling et al. [24] faces the threat of a session key leakage attack. Therefore, this paper mainly analyzes

and compares the identity authentication schemes, including sdronelig, SP-D2GCS [25], DroneSec [26], SENTINEL [29], and ACPBS-IoT [30], from the aspects of communication times, communication overhead, and time overhead.

The computational performance overhead of the sdronelig scheme is compared with that of existing UAV authentication schemes by theoretical calculation. On an Intel(R) Core(TM) i7-7700HQ CPU@2.80GHz 2.81 and GHz with a 16.0 GB environment to test the typical time cost of the operation that has a significant impact on the computational overhead of the authentication scheme, where the digital signature and verification are performed by the sign() function and verify() function in python keys.py, respectively, the digital certificate is generated by OpenSSL 1.1.1k based on the sha256ECDSA signature algorithm, and the certificate is verified by the verify_certificate() function in python cypto.py. The test method is to perform 1000 corresponding operations and take the average value, and the results are listed in Table 10.

Table 10. Time cost of typical cryptographic operations.

Type of Calculation	Description	Time Cost
Cal_{eccm}	elliptic curve point multiplication	0.421
Cal_{eccadd}	elliptic curve point addition	0.308
Cal_{sys}	symmetric encryption/decryption computation	0.014
Cal_{hash}	Message Digest Calculation (SHA256)	0.005
Cal_{sign}	Digital signature (secp256r1 curve)	0.771
Cal_{verif}	digital signature verification	2.775
Cal_{cert_verif}	certificate verification	2.945

The main computational overhead involved in each scheme is shown in Table 11.

Table 11. Comparison table of main calculation expenses of various schemes.

Certification Scheme	Computational Overhead
sdronelig	$3Cal_{eccm}+1Cal_{eccadd}+5Cal_{hash}+2Cal_{sys}$
SP-D2GCS [25]	$7Cal_{sys}+4Cal_{sign}+4Cal_{verif}+2Cal_{cert\_verif}+4Cal_{eccm}+11Cal_{hash}$
DroneSec [26]	$6Cal_{eccm}+2Cal_{eccadd}+8Cal_{hash}+2Cal_{sys}$
SENTINEL [29]	$2Cal_{sign}+2Cal_{verif}+2Cal_{cert\_verif}+2Cal_{sys}$
ACPBS-IoT [30]	$10Cal_{eccm}+3Cal_{eccadd}+18Cal_{hash}$

It can be seen from Table 12 that the sdronelig authentication scheme designed in this paper is superior to other similar UAV authentication schemes in all aspects. Compared with the identity authentication scheme between the UAV and the control station designed in Yongho et al. [25] and the SENTINEL [29], the communication overhead and the time overhead are both reduced by more than 50%, and compared with the identity authentication scheme designed in DroneSec [26], the calculation overhead, the number of communications, and the communication overhead of the authentication scheme are comprehensively reduced. Compared with ACPBS-IoT [30], the sdronelig scheme has the same communication times, but the communication overhead and calculation overhead are lower than ACPBS-IoT [30], which is more in line with the lightweight requirements of UAV communication.

Table 12. Comparison table of the main calculation expenses of various schemes.

Certification Scheme	Computational Overhead (ms)	Number of Communications (Times)	Communication Overhead (Bytes)
sdronelig	1.604	3	301
SP-D2GCS [25]	27.801	4	2411
DroneSec [26]	3.21	4	305
SENTINEL [29]	13.01	3	618
ACPBS-IoT [30]	5.224	3	2336

7. Conclusions

Authentication is one of the most critical factors in ensuring security, as it helps to prevent unauthorized access and data theft. Unfortunately, the existing UAV authentication schemes have some disadvantages, such as requiring special hardware, a lack of security, high overhead, and so on. In this article, an efficient and lightweight UAV authentication scheme called sdronelig is designed to solve the security threats faced by UAV communication networks. The scheme uses the ECDH algorithm to effectively resist the threat of session key disclosure. Without special hardware, the scheme realizes the mutual authentication of legitimate nodes in the UAV network and negotiates the security key, which can only be obtained by legitimate nodes. Compared with the existing UAV authentication scheme, the sdronelig scheme has reduced computation overhead, communication overhead, and communication times and meets the security and lightweight requirements of the UAV authentication scheme. At present, the authentication scheme designed in this paper only supports the key configuration based on the ground station UAV network, and it is difficult to deal with the authentication and encrypted communication between UAVs in a large-scale UAV cluster without a ground station. In future work, quantum communication technology and identification cryptographic algorithms should be used as references to further improve and adapt the communication security requirements in large-scale UAV cluster scenarios.

Author Contributions: Methodology, T.X., M.W. and J.H.; Formal analysis, T.X.; Investigation, L.G.; Resources, S.L. and Y.S.; Writing—original draft, T.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: We have presented the data in tabular form in the article.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

In order to verify the secrecy, confidentiality, and reachability of the session key, the software verification toolkit ProVerif 2.02 is utilized. Below is the simulation code for the proposed scheme in ProVerif2.02.

```
(* comments *)

(* define a public channel *)
free c : channel.
free readkey : channel [private].

(* define crypto primitive *)
(* symmetric encryption *)
type key.
fun key_to_bitstring(key) : bitstring [typeConverter].
fun senc(bitstring, key) : bitstring.
```

```
reduc forall m:bitstring, k:key; sdec(senc(m, k), k) = m.

(* mac *)
fun mac(bitstring, key) : bitstring.

type Bignum.
type Point.
fun point_to_bitstring(Point) : bitstring [typeConverter].
const G : Point [data].
fun ec_mul(Bignum, Point) : Point.
fun ec_add(Point, Point) : Point.
reduc forall n:Bignum, P:Point; de_ec_mul(n, ec_mul(n, P)) = P.
reduc forall A:Point, B:Point; de_ec_add(A, ec_add(A, B)) = B.
equation forall x:Bignum, y:Bignum; ec_mul(y, ec_mul(x, G)) = ec_mul(x, ec_mul(y, G)).
type skey.
type pkey.
fun skey_to_bignum(skey) : Bignum [typeConverter].
fun pkey_to_point(pkey) : Point [typeConverter].
fun pk(skey) : pkey.
fun sign(bitstring, skey) : bitstring.
reduc forall m : bitstring, sk : skey; checksign(m, sign(m, sk), pk(sk)) = true.
equation forall sk : skey; pkey_to_point(pk(sk)) = ec_mul(skey_to_bignum(sk), G).

event accept_g_station(key, bool).
event accept_UAV(key, bool).
event termi_g_station(key, bool).
event termi_UAV(key, bool).
event test(bool).

free safe_skey : skey [private].
free compromised_skey : skey [private].
free gs_skey : skey [private].
free secrecy : bitstring [private].

query attacker(secrecy).
query sk : key; event(termi_g_station(sk, false)).
query sk : key; event(termi_UAV(sk, false)).
query sk : key; inj-event(termi_g_station(sk, false)) ==> inj-event(accept_UAV(sk, false)).
query sk : key; inj-event(termi_UAV(sk, false)) ==> inj-event(accept_g_station(sk, false)).

let choose_UAV =
out(readkey, (pk(safe_skey), false));
out(readkey, (pk(compromised_skey), true)).

let g_station(id_s : bitstring,sk : skey) =
in(readkey, (P_d : pkey, compromised : bool));
(* read first gs msg *)
in(c, (id_d : bitstring, R_d : Point, mac_d : bitstring));
(* calculate vk *)

let point_vk = ec_add(R_d, ec_mul(skey_to_bignum(sk), pkey_to_point(P_d))) in
let key_to_bitstring(vk) = point_to_bitstring(point_vk) in
```

```
(* check mac *)
if mac_d = mac((id_d, point_to_bitstring(R_d)), vk) then
   new r_s : Bignum;
   let R_s = ec_mul(r_s, G) in
   let point_tmpkey = ec_mul(r_s, R_d) in
   let key_to_bitstring(tmpkey) = point_to_bitstring(point_tmpkey) in
   new k : key;
   new ktype : bitstring;
   let ek = senc(key_to_bitstring(k), tmpkey) in
   let mac_s = mac((id_s, point_to_bitstring(R_s), ek), vk) in
   (* reply *)
   event accept_g_station(k, compromised);

   out(c, (id_s, R_s, ek, mac_s));
   (* read second gs msg *)

   in(c, (id_d2 : bitstring, mac_d2 : bitstring));

if (id_d = id_d2) && (mac_d2 = mac((id_d2,k), tmpkey)) then
event termi_g_station(k, compromised);
   if not(compromised) then
      out(c, senc(secrecy, k)).

let UAV(id_d : bitstring, sk : skey, P_s : pkey, compromised : bool) =

new r_d : Bignum;
let R_d = ec_mul(r_d, G) in
let point_vk = ec_add(R_d, ec_mul(skey_to_bignum(sk), pkey_to_point(P_s))) in
let key_to_bitstring(vk) = point_to_bitstring(point_vk) in
let mac_d = mac((id_d, point_to_bitstring(R_d)), vk) in
out(c, (id_d, R_d, mac_d));

in(c, (id_s : bitstring, R_s : Point, ek:bitstring, mac_s : bitstring));
if mac_s = mac((id_s, point_to_bitstring(R_s), ek), vk) then
let point_tmpkey = ec_mul(r_d, R_s) in
let key_to_bitstring(tmpkey) = point_to_bitstring(point_tmpkey) in
let bit_k = sdec(ek, tmpkey) in
let key_to_bitstring(k) = bit_k in
event accept_UAV(k, compromised);
let mac_d2 = mac((id_d,k), tmpkey) in
out(c, (id_d, mac_d2));
event termi_UAV(k, compromised);
if not(compromised) then
      out(c, senc(secrecy, k)).

process
new id_s : bitstring;
new id_d1 : bitstring;
new id_d2 : bitstring;
let P_s = pk(gs_skey) in
let P_d1 = pk(safe_skey) in
out(c, P_s);
out(c, P_d1);
out(c, compromised_skey);
```

((!choose_UAV) | (!g_station(id_s, gs_skey)) | (!UAV(id_d1,safe_skey, P_s, false)) | (!UAV(id_d2, compromised_skey, P_s,true)))

(* ecc model test *)
(* process
new d_g : Bignum;
new d_u : Bignum;
new r_s : Bignum;
new r_d : Bignum;
let a = ec_mul(r_d, ec_mul(d_u, G)) in
out(c, (d_u, a));
in(c, P : Point);
if P = ec_mul(r_d, G) then
event ok *)

References

1. Li, W.; Yu, C.; Pu, W.; Zheng, Y. Security Threats and Countermeasures of Unmanned Aerial Vehicle Communications. *IEEE Commun. Stand. Mag.* **2021**, *5*, 41–47.
2. Khan, M.A.; Ullah, I.; Kumar, N.; Oubbati, O.S.; Qureshi, I.M.; Noor, F.; Khanzada, F.U. An Efficient and Secure Certificate-Based Access Control and Key Agreement Scheme for Flying Ad-Hoc Networks. *IEEE Trans. Veh. Technol.* **2021**, *70*, 4839–4851. [CrossRef]
3. Tan, Y.; Wang, J.; Liu, J.; Zhang, Y. Unmanned Systems Security: Models, Challenges, and Future Directions. *IEEE Netw.* **2020**, *34*, 291–297. [CrossRef]
4. Chen, C. *Research on Mutual Authentication Mechanism Based on ECC Algorithm*; Tsinghua University: Beijing, China, 2009.
5. Dolev, D.; Yao, A. On the security of public key protocols. *IEEE Trans. Inf. Theory* **1983**, *29*, 198–208. [CrossRef]
6. Edris, E.K.K.; Aiash, M.; Loo, L. Formal Verification of Authentication and Service Authorization Protocols in 5G-Enabled Device-to-Device Communications Using ProVerif. *Electronics* **2021**, *10*, 1608. [CrossRef]
7. Dong, R.; Wang, B.; Feng, D.; Cao, K.; Tian, J.; Cheng, T.; Diao, D. Physical layer secure transmission technology of UAV communication network. *J. Electron. Inform.* **2022**, *44*, 803–814.
8. Hu, L.; Bi, S.; Liu, Q.; Wu, J.; Yang, R.; Wang, H. Physical layer security algorithm for intelligent hypersurface-assisted UAV communication system based on reinforcement learning. *Chin. J. Electron. Inf. Sci.* **2022**, *44*, 2407–2415.
9. Maeng, S.J.; Yapıcı, Y.; Güvenç, I.; Bhuyan, A.; Dai, H. Precoder Design for Physical-Layer Security and Authentication in Massive MIMO UAV Communications. *IEEE Trans. Veh. Technol.* **2022**, *71*, 2949–2964. [CrossRef]
10. Wang, J.; Wang, X.; Gao, R.; Lei, C.; Feng, W.; Ge, N.; Jin, S.; Quek, T.Q.S. Physical Layer Security for UAV Communications: A Comprehensive Survey. *China Commun.* **2022**, *19*, 77–115. [CrossRef]
11. Allouch, A.; Cheikhrouhou, O.; Koubaa, A.; Khalgui, M.; Abbes, T. MAVSec: Securing the MAVLink Protocol for Ardupilot/PX4 Unmanned Aerial Systems. In Proceedings of the 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), Tangier, Morocco, 24–28 June 2019.
12. Kim, J.; Park, C.; Ahn, J.; Ko, Y.; Park, J.; Gallagher, J.C. Real-time UAV Sound Detection and Analysis System. In Proceedings of the 2017 IEEE Sensors Applications Symposium (SAS), Glassboro, NJ, USA, 13–15 March 2017.
13. Zhang, L.; Xu, J.; Obaidat, M.S.; Li, X.; Vijayakumar, P. PUF-based lightweight authentication and key agreement protocol for smart UAV networks. *IET Commun.* **2021**, *16*, 1142–1159. [CrossRef]
14. Tian, C.; Jiang, Q.; Li, T.; Zhang, J.; Xi, N.; Ma, J. Reliable PUF-based mutual authentication protocol for UAVs towards multi-domain environment. *Comput. Netw.* **2022**, *218*, 109421. [CrossRef]
15. Yoon, K.; Park, D.; Yim, Y.; Kim, K.; Yang, S.K.; Robinson, M. Security Authentication System Using Encrypted Channel on UAV Network. In Proceedings of the 2017 First IEEE International Conference on Robotic Computing (IRC), Taichung, Taiwan, 10–12 April 2017.
16. Srinivas, J.; Das, A.K.; Kumar, N.; Rodrigues, J.J.P.C. TCALAS: Temporal Credential-Based Anonymous Lightweight Authentication Scheme for Internet of Drones Environment. *IEEE Trans. Veh. Technol.* **2019**, *68*, 6903–6916. [CrossRef]
17. Ali, Z.; Chaudhry, S.A.; Ramzan, M.S.; Al-Turjman, F. Securing Smart City Surveillance: A Lightweight Authentication Mechanism for Unmanned Vehicles. *IEEE Access* **2020**, *8*, 43711–43724. [CrossRef]
18. Khalid, H.; Hashim, S.J.; Ahamed, S.M.S.; Hashim, F.; Chaudhary, M.A. Secure Real-time Data Access Using Two-Factor Authentication Scheme for the Internet of Drones. In Proceedings of the 2021 IEEE 19th Student Conference on Research and Development (SCOReD), Online. 23–25 November 2021.
19. Jan, S.U.; Abbasi, I.A.; Algarni, F. A Key Agreement Scheme for IoD Deployment Civilian Drone. *IEEE Access* **2021**, *9*, 149311–149321. [CrossRef]
20. Derhab, A.; Cheikhrouhou, O.; Allouch, A.; Koubaa, A.; Qureshi, B.; Ferrag, M.A.; Maglaras, L.; Khan, F.A. Internet of drones security: Taxonomies, open issues, and future directions. *Veh. Commun.* **2022**, *39*, 100552. [CrossRef]

21. Deebak, B.D.; Al-Turjman, F. A smart lightweight privacy preservation scheme for IoT-based UAV communication systems. *Comput. Commun.* **2020**, *162*, 102–117. [CrossRef]
22. Chen, Y.-J.; Wang, L.-C. Privacy Protection for Internet of Drones: A Network Coding Approach. *IEEE Internet Things J.* **2019**, *6*, 1719–1730. [CrossRef]
23. Won, J.; Seo, S.-H.; Bertino, E. Certificateless cryptographic protocols for efficient drone-based smart city applications. *IEEE Access* **2017**, *5*, 3721–3749. [CrossRef]
24. Chen, C.-L.; Deng, Y.-Y.; Weng, W.; Chen, C.-H.; Chiu, Y.-J.; Wu, C.-M. A Traceable and Privacy-Preserving Authentication for UAV Communication Control System. *Electronics* **2020**, *9*, 62. [CrossRef]
25. Ko, Y.; Kim, J.; Duguma, D.G.; Astillo, P.V.; You, I.; Pau, G. Drone Secure Communication Protocol for Future Sensitive Applications in Military Zone. *Sensors* **2021**, *21*, 2057. [CrossRef]
26. Jian, Q.; Chen, Z.; Wu, X. Authentication and key agreement protocol for UAV communication. *Comput. Sci.* **2022**, *49*, 306–313.
27. Alizadeh, J.; Safkhani, M.; Allahdadi, A. ISAKA: Improved Secure Authentication and Key Agreement protocol for WBAN. *Wirel. Pers. Commun.* **2022**, *126*, 2911–2935. [CrossRef]
28. Zhu, H.; Zhang, Y.; Yu, P.; Zhang, Z.; Wu, H.; Zhao, H. Key management and authentication protocol for UAV networks. *Eng. Sci. Technol.* **2019**, *51*, 158–166.
29. Cho, G.; Cho, J.; Hyun, S.; Kim, H. SENTINEL: A Secure and Efficient Authentication Framework for Unmanned Aerial Vehicles. *Appl. Sci.* **2020**, *10*, 3149. [CrossRef]
30. Bera, B.; Kumar Das, A.; Garg, S.; Piran, M.J.; Hossain, M.S. Access Control Protocol for Battlefield Surveillance in Drone-Assisted IoT Environment. *IEEE Internet Things J.* **2022**, *9*, 2708–2721. [CrossRef]