A Construction of Secure and Efficient Authenticated Key Exchange Protocol for Deploying Internet of Drones in Smart City

Dharminder Chaudhary(✉), Tanmay Soni, Soumyendra Singh, and Surisetty Mahesh Chandra Gupta

Department of Computer Science and Engineering, Amrita School of Computing
Amrita Vishwa Vidyapeetham, Chennai, India
manndharminder999@gmail.com

Abstract. The concept of a smart city is increasing because of the demand for intelligent drones. So, the Internet of Drones came into picture, providing several benefits/services for daily life. Services that IoD offers are monitoring, FANET (Flying Ad-Hoc Networks), management of any infrastructure, and IoT (Internet of Things). These services can be deployed in the smart city environment. Still, communication among drones is a significant concern. For communicating, drones use the insecure channel, and there is a risk of security threats while sending critical information. They are also prone to physical capture attacks because of their usage in an environment devoid of human beings. Regarding communication and computation, drones are resource constrained, so it is not feasible to implement public key cryptography because it requires more power to perform those actions. A recent protocol for the internet of drones is also analyzed in this article. Therefore, this article presents a lightweight authentication protocol that is reliable to the users and meets their demands. The performance analysis ensures the efficiency of the proposed protocol.

Keywords: Authentication · Key Agreement · Internet of Drones · Physical Unclonable Function

1 Introduction

The technologies are emerging with novel innovations being implemented simultaneously. Fields like IoT (Internet of Things), FANET (Flying Ad-hoc Networks), and 5G communication are also following the same trend, which led to the development of smart cities [1–5]. But there are certain challenges being faced by it like maintaining and storing a large amount of data. This data is collected by the sensors and IoT, which is later stored in their memory. One way to overcome this challenge is by combining UAV, FANET, and IoT into IoD

R. K. Challa et al. (Eds.): ICAIoT 2023, CCIS 1929, pp. 136–150, 2024.
https://doi.org/10.1007/978-3-031-48774-3_9

[6]. Each one plays their individual role even after getting combined. FANET, when used in UAVs, gives fast speed, low latency, and back-end services to the user. In drones also, when IoT is deployed, it gathers critical information from various difficult scenarios and carries that information forward with the help of FANET. So this led to the rapid demand for FANET-based IoD [7,8]. Their features include monitoring, surveillance, logistic transportation, and providing relief during critical rescue operations. The services which are available to use by IoD are possible because of the portability, flexibility, and rapid deployment. For example, if any natural calamity happens, the sensors present in the IoD can gather all the necessary information. Affected people can be located by their body temperature with the help of thermal sensors [9]. After this, medical assistance will be provided to them once the data collected by IoD is shared with them. Even after having many advantages still, there are certain issues that need to be addressed and resolved for better functioning of the IoD. FANET-based IoD uses public channels for sharing information which leads to the violation of privacy [10]. The data, which was gathered by the drones, can be used by the attacker for unethical activities if it is not secured properly. For example, an adversary can get the details of sensitive localities by taking pictures and recording videos and can even use them to transport illegal substances. Apart from this, adversaries can intrude into the privacy of someone else by clicking pictures with the help of a camera present in drones. If we talk of the extreme scenario, the adversary can gather information stored in the drone by capturing it and then pretend to be the captured drone using the information stored in it. Along with this, there is a boost in demand for services which is there in smart cities. And another challenge it faces is the lightweight property. IoD has a set of conditions or constraints like computation power and consumption of energy. This was the main objective for the evolution of a lightweight authentication scheme. It is difficult to perform highly complex computations, and any computation that a system is performing must be solved within a certain threshold time interval, or else the performance of the schema will be affected, and the desired result will not be achieved. So to ensure the reliable and efficient working of IoD in smart cities, secure and lightweight authentication and key agreement scheme is introduced [11]. For many years, researchers are proposing a schema that is secure and follows the condition of efficient Authentication Key Agreement (AKA) protocols for IoD so that they can be implemented in smart cities [12–14]. Some of them asserted that their scheme is reliable, efficient, untraceable, and capable of withstanding various attacks. Unfortunately, their assertion proved to be wrong when their schema faced security attacks. Elliptic Curve Cryptography and other existing AKA schemes can not be implemented in IoD-based smart cities because of their lightweight property.

2 Related Works

For many decades, several authentication and key agreement schemes have been developed in order to improve the security and privacy of IoT [15–17]. "Password-based single factor AKA scheme" was introduced by Lamport [18]. But Lamport

[18] was not successful because it was not resistant to password guessing attacks in offline mode because only password privacy and security was considered. So to improve this, "smart card based two-factor authentication and key agreement scheme and password" was introduced by Das [19]. But Das's [19] scheme also has a drawback which was explained by Nyang's and Lee's [20]. The drawback was prone to guessing passwords and capturing sensor node attacks. So the "Secure and efficient two-factor authentication and key agreement scheme" was introduced by Nyang and Lee [20] to overcome the drawback faced by Das's [19] scheme. To introduce the new feature of providing user anonymity and privacy, He et al. [21] proposed an "enhanced two-factor based authentication and key agreement scheme ." But He et al.'s [21] proposed a scheme was unable to establish the session key and also provide secure mutual authentication. This vulnerability was analyzed by Kumar and Lee [22]. So, it was concluded that two factors, AKA schemes for IoT [18–22] are prone to multiple security attacks. In previous years, several scholars have been introducing secure, and lightweight three-factor authentication and key agreement schemes based on biometric [23–27]. They were introducing these schemes so that IoT-enabled drones can overcome the vulnerability, security challenges and privacy challenges faced by the previous two factor-based authentication and key agreement scheme. The three-factor AKA-based scheme was introduced by Wazid et al. [27]. He described several security requirements for the different types of issues faced by IoD environments. But Wazid et al.'s scheme have one issue that it cannot guarantee perfect backward secrecy and independent aliasing. This was pointed out by Alladi et al. [24]. So he introduced the two stages of lightweight authentication and key agreement scheme for "software-defined network-based unmanned aerial vehicle." But Alladi et al. [24] have one issue, which was raised by Beebak et al. [25], that forgery, offline password guessing, and replay cannot be prevented. In addition to this, the confidentiality of the data and forward secrecy can not be guaranteed. To prevent this security issue "temporary login based anonymous lightweight three-factor authentication and key agreement scheme in the internet of drones," also known as TCALAS, was proposed by Srinivas et al. [26]. But Srinivas's scheme also faced one issue that it is prone to impersonation attacks and can also be traced easily. This was analyzed by Ali et al. [23]. Ali et al. [23] introduced an "enhanced authentication and key agreement scheme for the internet of drone-based smart city environments" to overcome the issue faced by the scheme developed by Srinivas et al. [26]. Still, for some reason, Ali et al. [23] scheme was vulnerable to forgery, server spoofing attacks, and session key disclosure. Similar to these schemes, others also tried to introduce new security features between the IoD and users, but those features have some vulnerabilities. The schemes [23–27] are still prone to security attacks in future. Recently, several public key cryptography-based authentication and key agreement schemes [28–31] have been introduced or proposed in the IoD based smart cities environment in order to enhance the level of security and privilege controls as compared to the previously defined schemes (three factor-based AKA schemes). "ECC based certificate less authentication and key agreement

scheme for the IoD-based smart city environment" was proposed by Won et al. [31]. But this scheme has some vulnerabilities, like it lacks anonymity of the user and formal security is not discussed. Another scheme, "Authentication and key agreement scheme to provide services in the internet of drone environments based on homomorphic encryption," was introduced by Cheon et al. [28]. But this scheme is prone to insider attacks and session key discloser. Another scheme [29] was proposed stating, "Secure and efficient authentication and key agreement framework for mobile sinks used in IoD environment based on bilinear pairing." But in several attacks like impersonation attacks and perfect forward secrecy is not guaranteed. In addition to these, this scheme cannot provide real-time services because of the high communication computation costs required by the bilinear pairing. This was pointed out by Nikooghadam et al. [30] who proposed a "Secure and lightweight authentication and key agreement scheme for the IoD-based smart city environments based on ECC," which is highly secure against several security compromised attacks. Later his scheme failed to be resistant against security attacks like replay, impersonation attacks, and insider and there was no mutual authentication between two or more devices. This vulnerability in the Nikooghadam et al. scheme [30] was discovered by Ali et al. [32]. These public key cryptography based authentication and key agreement schemes [28–31] can be prone to physical drone capture attacks. If it happens, then the adversary can extract all the sensitive information from it and impersonate the captured drone. These schemes are not made to solve complex problems because it requires high communication and computation cost.

3 Motivation and Contribution

The IoD is an emerging area for researchers, security and privacy are significant concerns for communication among drones. Many authentications and key exchange protocols have been proposed [23,26,27,30,31], but it was found that either they were not secure against possible attacks like password guessing, anonymity, Man in the middle [MITM], malicious insider, user/server impersonation, etc., or they are not efficient in terms of computation and communication. We have also studied a recent protocol [33] and found a vulnerability. According to the protocol to which we have referred,$\{id_i, rpw_i, r_i\}$ was submitted to the control server through a secure channel, but there is no meaning to send the random number r_i to the control server as it can be used by the attacker to retrieve the secret credential with certain attacks. Therefore, one need a secure and efficient authenticated key exchange mechanism for the IoD environment. This article proposes the required mechanism, which attains most of the security attributes. A performance analysis of the proposed protocol is also done, with relevant protocols, and found that the proposed protocol takes less computation cost. Therefore this protocol can be implemented for communication among IoD.

4 Threat Model

The most widely used threat model for finding the security of secure, lightweight authentication protocol for the IoD is Dolev and Yao [34] threat model. The actions which can be performed by the malicious attacker or adversary (MA) are discussed here. The Dolev and Yao model states that attackers can compromise, delete, eavesdrop, inject some codes and modify some of the data shared through the public medium. A malicious attacker can also perform a powerful analysis attack on smartphones by acting as a legal user [35]. This leads to the compromise of sensitive information present in the smartphone. Apart from this, the attacker can do capturing the physical device. After this, the login details can be extracted, and the attacker can behave like a legal user. The attacks performed by the attacker after getting the login details are forgery and impersonation attacks. Another scheme that is more secure and efficient than Dolev and Yao's(DY) threat model is Canetti and Krawczyk(CK), model [36]. It is also known as the CK threat model. The standard CK model is de facto for all AKA schemes. The CK threat model states that the attacker (MA) has all the capabilities mentioned in the DY model. But in addition to it, the adversary can also compromise sensitive information by performing attacks based on session hijacking. Also, a MA can perform the attack on CK [36] model like an ephemeral secret leakage attack. So it is essential for any scheme that it does not reveal the data even after the session hijacking is performed. The scheme must be resistible enough to protect the data of other devices connected to it.

5 Physical Unclonable Functions

When we talk about security, there are hardware devices like sensors and the IoD. To protect them from the malicious adversary or attacker, physical unclonable functions(PUF) [37] is used. Generally, PUF produces single unique output for the input which is given to it. For example, the fingerprint sensor present in the devices uses PUF. The unique feature of the PUF is that the secret key is not saved, and public key cryptography is not used for authentication purposes. Apart from this, making the exact copy of the previously known PUF is difficult because they are designed by nanoscale variation at the time of production of integrated circuit chips. PUF are durable, unique and tamperproof. They protect the devices connected from side-channel attacks and cloning. The mathematical way of representing PUF is $O = PUF(c)$, where O is output; PUF is a physical unclonable function, and c is a challenge given to it. Properties of PUF are (1) architecture of PUF decides the output, (2) produced output is unclonable and unique, and they are easy to implement on any device. In case there is any modification of PUF during manufacturing, the output will also be affected. The user and device authenticity is verified by a PUF before the connection is established [38]. If we implement these features into our scheme, it will result in an efficient and durable model.

6 Network Model

The internet of the drone-enabled smart city will have the network model. The suggested scheme has three major structural elements: control server(cs), drone(d), and mobile user(mu). A brief description of each in detail is as follows:

1. Control Server(cs): As the name suggests, it refers to the ground station server. They act as a medium between the drone and the mobile user. It allows the mobile user to communicate with the drones. Apart from this, mobile users can monitor the drones provided that the mobile users must be authorized. To authorize the entity's control server assigns the credentials to mobile users and drones and registers them. One can say that the control server is the link between mobile users and drones. The mobile user and the drone can also mutually authenticate themself in the public channel through which they are connected. This authentication is possible with the help of the control server only. The database of the control server is not accessible by the malicious adversary(MA). So the data stored in the database is secured.
2. Mobile user(mu): The users who carry mobile devices like smartphones, tablets, etc., are referred to as mobile users. In the registration phase, the credential is assigned to the mobile user by the control server. Once the credentials are given, mutual authentication is performed for mobile users and drones. After this, the session key is established between them for secure communication and data sharing.
3. Drone(d): Similar to the mobile user, in the registration phase, the credential is assigned to the drone by the control server. Once the credentials are given, they are eligible for deployment in flying areas. The drone which is deployed is controlled by the control server in order to send data collected by it using the sensors. The data, which is collected, is then sent to mobile users.

7 Proposed Secure and Efficient Authenticated Key Exchange Protocol for Deploying Internet of Drones in Smart City

This section discusses the proposed authenticated key exchange for deploying the IoD in the smart city. The protocol is divided into four phases (i) initialization, (ii) drone registration, (iii) user registration, and (iv) authentication and key agreement phases, respectively. Table 1 shows all the most important notations and terminologies.

7.1 Initialization

In this phase, the public parameters of the system, such as physical unclonable function PUF(.) and fuzzy extractor functions like generator gen(.) and reproduction Rep(.), are issued. In addition, the control server also chooses the Z_p and msk, which belongs to Z_p. The control server does certain pre-assignment

Table 1. Notations

Symbol	meaning
mu_i	Mobile user
d_j	Drone
cs	control server
bio_i	biometric of mobile user
id_i, pw_i	identity and password of mobile user
did_j	identity of drone
r_1, r_2, r_3	random nonces
t_i	timestamp
sk	session key between mobile user and drone
msk	maskter key of control server
$h()$	hash function
$PUF()$	physical unclonable function
$\oplus$	XOR operation
$\|\|$	Concate operation

operations, like the secret credentials assigned to a drone for registering and authenticating before they can operate in their flying areas. The Drone identity (did) is chosen for each drone by the control server. Then the drone identity (did) is sent to the respective drones. The drones save their identity in the secure database. Once initialized, the system goes to the next phase of registration.

7.2 Drone Registration

After registering, the drones receive their secret credentials. Drones use the credentials for authentication purposes. Drone registration is elaborated in three steps as follows:

1. DRP 1: The drone d_j selects its did_j and arbitrary number n_j. Then drone d_j submit its identity and random number as a single entity $\{n_j, did_j\}$ to the control server through the secure transmission channel.
2. DRP 2: After receiving the message, the control server checks if $did_j^* = did_j$. If true, then a random challenge set c_j is selected by the control server. The response $Res_j = PUF(c_j)$ is calculated with that challenge c_j. Two element set r_j and δ_j are calculated when the response is passed to the PUF $Gen(.)$ function with a condition that $(r_j, \delta_j) = (Gen(Res_j))$. Then, the control server calculates $z_j = h(did_j||msk)$, $n_j = z_j \oplus h(did_j||n_j)$ and $e_j = \delta_j \oplus h(z_j||n_j||did_j)$, which sent $\{n_j, e_j\}$ to the drone. At last, the control server stores $\{z_j, (c_j, r_j)\}$ as a single entity in its memory.
3. DRP 3: Once the message is received, the secret credential $\{n_j, e_j\}$ is stored by the drone in its memory.

7.3 Mobile User Registration

After registering, the mobile users mu_i receive their secret credentials. Then mobile users mu_i use the credentials for authentication. Mobile user registration is described briefly in 3 steps mentioned below:

1. URP:1 mu_i represents mobile user, selects a mobile user identity id_i, password pw_i and random number r_i. After this, the mobile user mu_i computes a random password $rpw_i = h(pw_i||r_i)$and sends identity and random password $\{id_i, rpw_i\}$as a single entity to the control server through the secure transmission channel.
2. URP:2 After receiving the message, the control server calculates a random identity $rid_i = h(id_i||rpw_i||msk_i)$, $x_i = h(msk_i||rid_i||rpw_i)$ and saves $\{rid_i, x_i\}$ as a single entity in its memory. After this, cs the fetches did_j and submit $\{did_j, rid_i, x_i\}$ to mobile user through a secure medium.
3. URP:3 Once the message is received, the mobile user calculates $\beta_i^* = \beta \oplus h(id_i||pw_i)$, $x_i^* = x_i \oplus h(id_i||pw_i||r_i)$, $rid_i^* = rid_i \oplus h(pw_i||id_i||rpw_i)$, $did_j^* = did_j \oplus h(id_i||rpw_i||pw_i)$, and $c_i^* = h(rid_i||pw_i||x_i||r_i)$. Then, gateway gw_i changes $\{rid_i^*, x_i^*\}$ with $\{rid_i, x_i\}$ after that it saves $\{\beta_i^*, c_i^*, did_i^*\}$ in the mobile user device which is connected to it.

7.4 Authentication and Key Agreement Process

In the Authentication process, Control Server authenticates the mobile user and drone to establish a session key. A detailed description of AKA process of our proposed model is stated below:

1. AKP-1: A user enters its secret id_i and pw_i in the mobile devices. r_i=, rpw_i = h$(pw_i \parallel r_i)$, $x_i = x_i^* \oplus$ h$(id_i \| pw_i \| r_i)$, $rid_i = rid_i^* \oplus$ h$(pw_i \parallel id_i \|$ $rpw_i)$, $did_j = did_j^* \oplus$ h$(id_i \parallel rpw_i \parallel pw_i)$, and c_i^* =h$(rid_i \parallel pw_i \parallel x_i \parallel$ $r_i)$, and verifies $c_i^* \overset{?}{=} c_i$. If its false then the session is aborted by mu_i if not, chooses a arbitrary numpty r_1 and a timestamp t_1, and calculates m_1=$(r_1 \| did_j) \oplus$h$(rid_i \| x_i \| t_1)$ and $auth_u s$=h$(rid_i \| r_1 \| x_i \| t_1)$, and over a public channel the message $\{m_1, auth_u s, rid_i, t_1\}$ is sent to CS.
2. AKP-2: After receiving the messages, a timestamp t_2 is generated, and freshness of $|t_2 - t_1| \leq \triangle t$ is verified and where maximum time delay for transmission is denoted by $\triangle t$ and message reception time is denoted as t_2. If the date and time of the message are valid, Control Server calculates $(r_1 \parallel did_j)$ = $m_1 \oplus$h$(rid_i \parallel x_i \parallel t_1)$ and $auth_{us}^* = auth_{us}$. If it's true, the control server fetches $(c_j, r_j) \leftarrow did_j$, and chooses an arbitrary numpty r_2. Then, control server calculates z_j=h$(did_j \parallel z_j \parallel t_2)$ and $auth_{sd}$ = h$(did_j \parallel r_2 \parallel z_j \parallel \| r_j \parallel t_2)$ and sends $\{c_j, m_2, auth_{sd}, t_2\}$ to the drone.
3. AKP-3: Once the message is received, current timestamp t_3 is selected by d_j and validity of $|t_3 - t_2| \leq \triangle t$ is checked by d_j. If its true, d_j retrives $\{n_j, e_j\}$ in the memory and calculates $z_j = n_j \oplus$h$(did_j \parallel b_j)$ and $\delta_j = e_j \oplus$h$(z_j \parallel b_j \parallel$ $did_j)$. After this drone computes r_j=rep(PUF(c_j),δ_j),$(r_1 \parallel r_2)$=$m_2 \oplus$h$(did_j \parallel$

$r_j \parallel z_j \parallel t_2$) and $auth^*_{sd}$=h($did_j \parallel r_2 \parallel r_3 \parallel r_j \parallel t_3$) and $auth_{du}$=h($r_1 \parallel r_3 \parallel r_j \parallel did_j \parallel sk$). Finally, Over a public channel message $m_3, auth_{ds}, auth_{du}, t_3$ to control server by d_j.

4. AKP-4: After receiving the messages, The timestamp t_4 is generated by CS and freshness of $|t_4 - t_3| \leq \triangle t$ is verified. If it is valid, CS calculates r_3=m_3⊕h($r_j \parallel r_2 \parallel t_3$) and $auth^*_{ds}$=h($did_j \parallel r_2 \parallel r_3 \parallel r_j \parallel t_3$) and verifies if $auth^*_{ds} \overset{?}{=} auth_{ds}$. Whenever affliction is valid, control server calculates m_4=($r_2 \parallel r_3 \parallel r_j$⊕)h($rid_i \parallel did_j \parallel r_1 \parallel x_i \parallel t_4$) and $auth_{su}$=h($rid_i \parallel r_1 \parallel r_2 \parallel x_i$). Finally, CS sends $m_4, auth_{du}, auth_{su}, t_4$ to mu_i.
5. AKP-5: Upon receiving the messages, mu_i a timestamp t_5 is selected and $|t_5 - t_4| \leq \triangle t$ is verified. If they are same then mu_i calculates ($r_2 \parallel r_3 \parallel r_j$)=$m_4$⊕h($rid_i \parallel r_1 \parallel r_2 \parallel x_i$) and validate whether $auth^*_{su} \overset{?}{=} auth_{su}$. If the validation fails mu_i closes the current session alternatively mu_i calculates sk = h($r_1 \parallel r_3 \parallel r_j$) and $auth^*_{ds} \overset{?}{=} auth_{ds}$. The session key sk will be established successfully if its valid and mobile user and drone are authenticated mutually.

8 Security Analysis

A security analysis is done to prove the security of our proposed scheme. To analyze the security of our proposed model or scheme, the informal security analysis is discussed. In this, it is to prove that the model is immune to specific attacks like revealing anonymity, disclosing session key agreements, etc.

8.1 Impersonation Attacks

This attack happens when a malicious attacker or adversary tries to impersonate a legal or authorized user mu_i by intercepting the communication sent through the public medium. But while performing this attack, the malicious adversary must know the $\{m_1, auth_{us}, rid_i, t_1\}$ and $\{m_4, auth_{du}, auth_{su}, t_4\}$. So an attacker can't get the messages because the attacker fails to get the arbitrary numpty r_1 and private credential x_i. At last, it can be said that the model is immune to impersonation attacks.

8.2 Replay Attack

Suppose the adversary intercepts $\{m_1,\ auth_{us},\ rid_i,\ t_1\}$, $\{c_j,\ m_2,\ auth_{sd},\ t_2\}$, $\{m_3,\ auth_{ds},\ auth_{du},\ t_3\}$, and $\{m_4,\ auth_{du},\ auth_{su},\ t_4\}$ are sent during the authentication and key agreement phase. Whenever a malicious attacker tries to resent the previously sent messages then our model will check for the current timestamp present on it or not. Here, the messages which are sent have private credentials x_i, and random nonces $\{r_1, r_2, r_3\}$. So our model is immune to replay attack.

8.3 Physical Capture Attack on Drones

Suppose a malicious attacker captures some drones to extract all the sensitive information $\{n_j, e_j\}$ where $n_j = z_j \oplus h(did_j||b_j)$ and $e_j = \delta_j \oplus h(z_j||b_j||did_j)$ present in its memory. But, a malicious adversary cannot calculate the common session key which was established because the adversary does not know the drone's arbitrary number b_j and arbitrary numpty r_2. And each drone has a unique and independent random number and random nonce. This happens because challenge and response (c_j, r_j) are randomly formed. So it is very difficult to find the common session key and it can be concluded that our scheme is immune to physical capture attacks.

8.4 Disclosed Session Key Attack

If the malicious attacker gets the private credentials $\{x_i^*, rid_i^*, c_i^*, did_j^*, \beta_i^*, \}$ after performing a stolen password attack to imitate a legal user mu_i. But the attacker must have random nonces $\{r_1, r_2\}$, response $\{r_j\}$ to get the session key $sk = h(r_1, r_3, r_j)$. But an attacker can't get the random nonces because all the nonces are masked with private credentials $\{x_i, z_j\}$. In addition to it, the attacker also does not know the physically unclonable function's private parameters δ_j. So our model is immune to the disclosed session key attacks.

8.5 Offline Password Guessing Attack

If the malicious attacker ma gets the private credentials after performing the attack mentioned in the threat model section. So malicious attackers can try all possibilities to get the real password $pw_i$of the legal user mu_i. But, the actual password is hashed with the random number and stored as a random password $rpw_i = h(pw_i||r_i)$. So it is difficult to get pw_i if the arbitrary number r_i is not known to the adversary. So the model is immune to offline password-guessing attacks.

8.6 Man-in-the-middle Attack

As mentioned in the threat model section, a malicious attacker can eavesdrop on $\{m_1,\ auth_{us},\ rid_i,\ t_1\}$, $\{c_j,\ m_2,\ auth_{sd},\ t_2\}$, $\{m_3,\ auth_{ds},\ auth_{du},\ t_3\}$, and $\{m_4,\ auth_{du},\ auth_{su},\ t_4\}$ sent through public medium and tries to perform a man-in-the-middle attack. But the attacker can not get the authentication and confirmation messages because the arbitrary number $\{r_1, r_2, r_3\}$ and secret credentials $\{x_i, z_j\}$ have masked them. In addition to it, a malicious adversary cannot get the session key $sk = h(r_1||r_2||r_3)$ without having the arbitrary numpty $\{r_1, r_2, r_3\}$ and physical unclonable function response parameter r_j. So our model is immune to a man-in-the-middle attack.

8.7 Ephemeral Secret Leakage Attack

As mentioned in the Canetti Krawczyk model, if the malicious adversary ma gets only the private credentials and session states from the other features as mentioned in Dolev and Yao model. Even if the long-term keys $\{r_i, b_j\}$ are known to the attacker, the session key sk is still unknown because the real identity $\{id_i, did_j\}$and secret value of physical unclonable functions $\alpha_i$are not known to the attacker. While the other case can be if an attacker gets the short-term keys $\{r_1, r_2, r_3\}$ still the session key sk cannot be revealed because the private credentials $\{x_i, z_j\}$ and secret parameters of the physically unclonable function $\{r_j\}$ are not known to the attacker. So our model is immune to the ephemeral secret leakage attack.

8.8 Anonymity

If malicious adversary ma eavesdrop on the messages, send them during the authentication and key agreement phase. Still, it is difficult for the malicious adversary to get the real identity $\{id_i, did_j\}$ if the master key msk and masked password rpw_i are not known. This happens because $\{id_i, did_j\}$ of mobile user and drone are stored as $rid_i = h(id_i||rpw_i||msk)$ and $did_j^* = did_j \oplus h(id_i||rpw_i||pw_i)$. So our model offers anonymity.

8.9 Mutual Authentication

Our model mutually authenticates all the connected devices. The control server gets the login request $\{m_1, auth_{us}, rid_i, t_1\}$ from the mobile user and it checks $auth_{us}^* \overset{?}{=} h(rid_i||r_1||x_i||t_1)$. If it's true, then the control server authenticates the mobile user. Similarly, the drone also verifies the authentication request message $\{c_j, m_2, auth_{ds}, t_2\}$ is sent by the control server and checks authentication $auth_{sd}^* \overset{?}{=} h(did_j||t_2||r_j||z_j||r_2)$. If it's true, then the drone authenticates the control server. After this, a confirmation message $\{m_3, auth_{ds}, auth_{du}, t_3\}$ reaches to control server from the drone. Then, the control server checks if it's true or not $auth_{ds}^* \overset{?}{=} h(did_j||r_2||r_3||r_j||t_3)$. If it's true, control authenticates the drone. Now the authentication confirmation messages $\{m_4, auth_{du}, auth_{su}, t_4\}$are sent to mobile users from drones and control servers. If $auth_{su}^* \overset{?}{=} h(rid_j||r_1||r_2||x_i)$ and $auth_{du}^* \overset{?}{=} h(r_1||r_2||r_j||did_j||sk)$ are true, then the mobile user authenticates the drone and control server. As a result, the model provides mutual authentication to all the devices connected.

9 Performance Analysis

The proposed scheme's authentication and key agreement computation overhead and communication overhead are compared with the previously proposed schemes, as mentioned in related works.

9.1 Computation Time

The overhead computation cost of the proposed scheme has been analysed by comparing it with the previously published schemes [23,26,27,30,31,33]. This comparison is made at the authentication and key agreement phase. For comparison, the testbed experiment results will be used. The measurement of cost is done based on the computation time defined for the cryptographic primitives. The average time required by the cryptographic primitives, particularly for the control server, is represented by the Table 2.

Table 2. Computation Cost

Notations	Scheme	mu_i	cs	d_j	Total Cost
$[A]$	[27]	$t_{fe} + 16t_h \approx 7.792$ ms	$8t_h \approx 0.44$ ms	$7t_h \approx 2.163$ ms	10.395 ms
$[B]$	[26]	$t_{fe} + 14t_h \approx 7.174$ ms	$9t_h \approx 0.495$ ms	$7t_h \approx 2.163$ ms	9.832 ms
$[C]$	[23]	$t_{fe} + 10t_h \approx 5.938$ ms	$3t_{sed} + 7t_h \approx 0.388$ ms	$7t_h \approx 2.163$ ms	8.489 ms
$[D]$	[31]	$5t_{ecpm} + 5t_h \approx 15.785$ ms	–	$4t_{ecpm} + 2t_h \approx 12.01$ ms	27.795 ms
$[E]$	[30]	$2t_{ecpm} + 6t_h \approx 7.55$ ms	$8t_h \approx 0.44$ ms	$2t_{ecpm} + 5t_h \approx 7.241$ ms	15.231 ms
$[F]$	[33]	$t_{fe} + 12t_h \approx 6.556$ ms	$9t_h \approx 0.495$ ms	$t_{fe} + 8t_h \approx 5.32$ ms	12.371 ms
$[G]$	$ProposedProtocol$	$t_{fe} + 16t_h \approx 7.692$ ms	$8t_h \approx 0.43$ ms	$7t_h \approx 2.162$ ms	10.284 ms

In such cases, t_{bp} , t_{ecpm}, t_{fe}, t_h and t_s is considered. Now the average time required by the cryptographic primitives, particularly for the drones and the mobile user, is represented in Table 2. In such cases, t_{bp} , t_{ecpm}, t_{fe}, t_h and t_s is considered. The result of our computation cost when compared with the previously defined schemes, is represented in Table 2 and Fig. 1. After seeing the result, it can be concluded that the proposed scheme [G] gives less computation time when compared with other schemes [27] denoted by [A], [26] [B], [23] denoted by [C], [31] denoted by [D], [30] [E], and [33] denoted by [F] in the Fig. 1.

9.2 Communication Cost

For finding the communication cost, several factors need to be considered. For example, timestamp size, ciphertext and plaintext of the private key algorithm, any arbitrary number, identity, $h(.)$, and various elliptic curve points like 32, 64, 128, 160, and 320 bits. The message created and sent by the mobile user in the secure lightweight authentication protocol internet of drones is $\{m1, auth_{us}, rid_i, t_1\}$. The total overhead is 512 bits. Similarly, control server also creates some message $\{c_j, m_2 auth_{sd}, t_2\}$ and $\{m_4, auth_{du}, auth_{su}, t_4\}$ and sent them. The total overhead of the control server is 512 bits. Likewise drone also create a message $\{m_3 auth_{ds}, auth_{du}, t_3\}$. The total overhead of the drone is 512 bits. The result of our communication cost compared with the previously defined schemes is represented by the Table 3.

The total overhead of communication cost of the proposed scheme is 2048 bits. After seeing the result, it can be concluded that the proposed scheme gives less communication time when compared with other schemes [23,26,27,30,31].

In addition to it, the security and efficiency are also better than other schemes. In Fig. 1, an analysis of the proposed protocol with relevant ones is shown.

Table 3. Communication cost

Schemes	Message 1	Message 2	Message 3	Message 4	Total Cost
[27]	672 bits	512 bits	512 bits	- bits	1696 bits
[26]	672 bits	512 bits	352 bits	- bits	1536 bits
[23]	480 bits	672 bits	512 bits	- bits	1664 bits
[31]	1952 bits	- bits	- bits	- bits	1952 bits
[30]	832 bits	992 bits	512 bits	- bits	2336 bits
[33]	512 bits	512 bits	512 bits	512 bits	2048 bits
proposed protocol	512 bits	512 bits	512 bits	512 bits	2048 bits

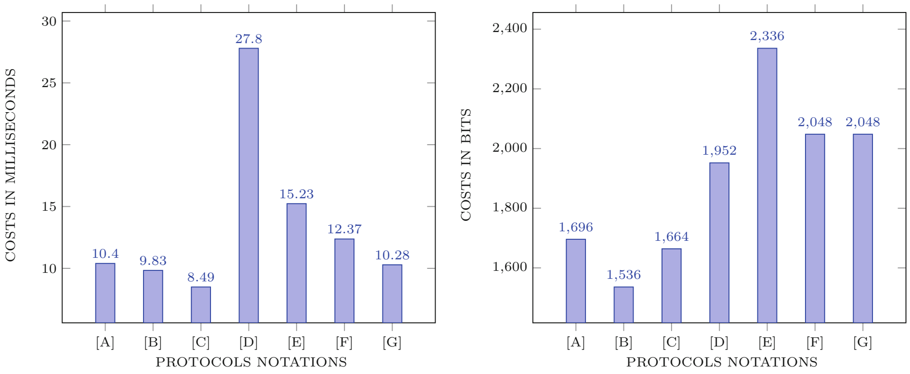

Fig. 1. Illustration of Computations and Communications Costs of Protocols

10 Conclusion

Many authentications, and key exchange protocols were studied, but it was found that either they were not secure against possible attacks like password guessing, anonymity, Man in the middle, malicious insider, user/server impersonation, etc., or they are not efficient in terms of computation and communication. A recent study on protocol [33] was studied and a vulnerability was found. According to the protocol to which we have referred,$\{id_i, rpw_i, r_i\}$ was submitted to the control server through a secure channel but there is no meaning to send the random number r_i to the control server as the attacker can use it to retrieve the secret credential with certain attacks. Therefore, we have designed a secure and efficient authenticated key exchange mechanism for the internet of drones environment. We have also done a performance analysis of the proposed protocol, with relevant protocols, and found that the proposed protocol takes less computation cost. Therefore this protocol can be implemented for communication among the Internet of Drones.

References

1. Das, A.K., Bera, B., Wazid, M., Jamal, S.S., Park, Y.: igcacs-iod: an improved certificate-enabled generic access control scheme for internet of drones deployment. IEEE Access **9**, 87024–87048 (2021)
2. Khan, M.A., et al.: An efficient and secure certificate-based access control and key agreement scheme for flying ad-hoc networks. IEEE Trans. Veh. Technol. **70**(5), 4839–4851 (2021)
3. Li, X., Jianwei Niu, Md., Bhuiyan, Z.A., Fan, W., Karuppiah, M., Kumari, S.: A robust ECC-based provable secure authentication protocol with privacy preserving for industrial internet of things. IEEE Trans. Ind. Inf. **14**(8), 3599–3609 (2017)
4. Mandal, S., Bera, B., Sutrala, A.K., Das, A.K., Choo, K.K.R., Park, Y.: Certificateless-signcryption-based three-factor user access control scheme for IoT environment. IEEE Internet Things J. **7**(4), 3184–3197 (2020)
5. Yu, S., Lee, J., Park, K., Das, A.K., Park, Y.: IoV-SMAP: secure and efficient message authentication protocol for IoV in smart city environment. IEEE Access **8**, 167875–167886 (2020)
6. Long, T., Ozger, M., Cetinkaya, O., Akan, O.B.: Energy neutral internet of drones. IEEE Commun. Maga. **56**(1), 22–28 (2018)
7. Boccadoro, P., Striccoli, D., Grieco, L.A.: An extensive survey on the Internet of Drones. Ad Hoc Netw. **122**, 102600 (2021)
8. Gharibi, M., Boutaba, R., Waslander, S.L.: Internet of drones. IEEE Access **4**, 1148–1162 (2016)
9. Mishra, B., Garg, D., Narang, P., Mishra, V.: Drone-surveillance for search and rescue in natural disaster. Comput. Commun. **156**, 1–10 (2020)
10. Yahuza, M., et al.: Internet of drones security and privacy issues: taxonomy and open challenges. IEEE Access **9**, 57243–57270 (2021)
11. Wazid, M., Das, A.K., Lee, J.K.: Authentication protocols for the internet of drones: taxonomy, analysis and future directions. J. Ambient Intell. Human. Comput. 1–10 (2018)
12. Chaudhry, S.A., Yahya, K., Karuppiah, M., Kharel, R., Bashir, A.K., Zikria, Y.B.: GCACS-IoD: a certificate based generic access control scheme for Internet of drones. Comput. Netw. **191**, 107999 (2021)
13. Cho, G., Cho, J., Hyun, S., Kim, H.: Sentinel: a secure and efficient authentication framework for unmanned aerial vehicles. Appl. Sci. **10**(9), 3149 (2020)
14. Zhang, Y., He, D., Li, L., Chen, B.: A lightweight authentication and key agreement scheme for internet of drones. Comput. Commun. **154**, 455–464 (2020)
15. Gope, P., Hwang, T.: A realistic lightweight anonymous authentication protocol for securing real-time application data access in wireless sensor networks. IEEE Trans. Ind. Electron. **63**(11), 7124–7132 (2016)
16. Park, K., et al.: LAKS-NVT: provably secure and lightweight authentication and key agreement scheme without verification table in medical internet of things. IEEE Access **8**, 119387–119404 (2020)
17. Shen, J., Zhou, T., Wei, F., Sun, X., Xiang, Y.: Privacy-preserving and lightweight key agreement protocol for v2g in the social internet of things. IEEE Internet Things J. **5**(4), 2526–2536 (2017)
18. Lamport, L.: Password authentication with insecure communication. Commun. ACM **24**(11), 770–772 (1981)
19. Manik Lal Das: Two-factor user authentication in wireless sensor networks. IEEE Trans. Wirel. Commun. **8**(3), 1086–1090 (2009)

20. Nyang, D.H., Lee, M.K.: Improvement of das's two-factor authentication protocol in wireless sensor networks. Cryptology EPrint Archive (2009)
21. He, D., Gao, Y., Chan, S., Chen, C., Jiajun, B.: An enhanced two-factor user authentication scheme in wireless sensor networks. Ad Hoc Sens. Wirel. Netw. **10**(4), 361–371 (2010)
22. Kumar, P., Lee, H.J.: Cryptanalysis on two user authentication protocols using smart card for wireless sensor networks. In: 2011 Wireless Advanced, pp. 241–245. IEEE (2011)
23. Ali, Z., Chaudhry, S.A., Ramzan, M.S., Al-Turjman, F.: Securing smart city surveillance: a lightweight authentication mechanism for unmanned vehicles. IEEE Access **8**, 43711–43724 (2020)
24. Alladi, T., Chamola, V., Kumar, N., et al.: Parth: a two-stage lightweight mutual authentication protocol for UAV surveillance networks. Comput. Commun. **160**, 81–90 (2020)
25. Bakkiam David Deebak and Fadi Al-Turjman: A smart lightweight privacy preservation scheme for IoT-based UAV communication systems. Comput. Commun. **162**, 102–117 (2020)
26. Srinivas, J., Das, A.K., Kumar, N., Rodrigues, J.J.: TCALAS: temporal credential-based anonymous lightweight authentication scheme for Internet of drones environment. IEEE Trans. Veh. Technol. **68**(7), 6903–6916 (2019)
27. Wazid, M., Das, A.K., Kumar, N., Vasilakos, A.V., Rodrigues, J.J.: Design and analysis of secure lightweight remote user authentication and key agreement scheme in internet of drones deployment. IEEE Internet Things J. **6**(2), 3572–3584 (2018)
28. Cheon, J.H., et al.: Toward a secure drone system: flying with real-time homomorphic authenticated encryption. IEEE Access **6**, 24325–24339 (2018)
29. Yoney Kirsal Ever: A secure authentication scheme framework for mobile-sinks used in the internet of drones applications. Comput. Commun. **155**, 143–149 (2020)
30. Nikooghadam, M., Amintoosi, H., Islam, S.H., Moghadam, M.F.: A provably secure and lightweight authentication scheme for Internet of Drones for smart city surveillance. J. Syst. Arch. **115**, 101955 (2021)
31. Won, J., Seo, S.-H., Bertino, E.: Certificateless cryptographic protocols for efficient drone-based smart city applications. IEEE Access **5**, 3721–3749 (2017)
32. Ali, Z., Alzahrani, B.A., Barnawi, A., Al-Barakati, A., Vijayakumar, P., Chaudhry, S.A.: TC-PSLAP: temporal credential-based provably secure and lightweight authentication protocol for IoT-enabled drone environments. Secur. Commun. Netw. **2021**, 1–10 (2021)
33. Yu, S., Das, A.K., Park, Y., Lorenz, P.: SLAP-IoD: secure and lightweight authentication protocol using physical unclonable functions for internet of drones in smart city environments. IEEE Trans. Veh. Technol. **71**(10), 10374–10388 (2022)
34. Dolev, D., Yao, A.: On the security of public key protocols. IEEE Trans. Inf. Theory **29**(2), 198–208 (1983)
35. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 388–397. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48405-1_25
36. Canetti, R., Krawczyk, H.: Universally composable notions of key exchange and secure channels. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 337–351. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-46035-7_22
37. Aman, M.N., Chua, K.C., Sikdar, B.: Mutual authentication in iot systems using physical unclonable functions. IEEE Internet Things J. **4**(5), 1327–1340 (2017)
38. Gao, Y., Al-Sarawi, S.F., Abbott, D.: Physical unclonable functions. Nat. Electron. **3**(2), 81–91 (2020)