GCACS-IoD: A certificate based generic access control scheme for Internet of drones

Shehzad Ashraf Chaudhry[a], Khalid Yahya[b], Marimuthu Karuppiah[c], Rupak Kharel[d], Ali Kashif Bashir[d], Yousaf bin Zikria[e,*]

[a]*Department of Computer Engineering, Faculty of Engineering and Architecture, Istanbul Gelisim University, 34310 Istanbul, Turkey*
[b]*Department of Mechatronics Engineering, Faculty of Engineering and Architecture, Istanbul Gelisim University, 34310 İstanbul, Turkey*
[c]*Department of Computer Science and Engineering, SRM Institute of Science and Technology, Delhi NCR Campus, Modinagar, Ghaziabad, Uttar Pradesh 201204, India.*
[d]*Department of Computing and Mathematics,Manchester Metropolitan University, M15 6BH Manchester, UK*
[e]*Department of Information and Communication Engineering, Yeungnam University, Gyeongsan, 38541, South Korea*

Abstract

Internet of drones (IoD) has gained significant importance in recent times due to its applications in several critical domains ranging from commercial to defense and rescue operations. With several drones flying in different zones to carry out specified tasks, the IoD can be beneficial to gather the real time data for interpretation by the users. However, the data access is carried out through an open channel and battery operated drones. Therefore, the drones' security and privacy are crucial for accomplishing mission-critical, safety-critical, or surveillance operations. In 2020, Bera et al. presented a certificate based access control scheme for securing the IoD access and argued the scheme's security through formal and informal methods. However, the analysis presented in this paper shows that the scheme of Bera et al. does not provide anonymity and is insecure against multiple threats, including drone impersonation, the man in the middle, and replay attacks. We then designed a generic certificate based access control scheme to provide inter-drone and drone to ground station access control/authentication scheme in the IoD domain (GCACS-IoD). The GCACS-IoD is provably secure against the known attacks and provides anonymity. GCACS-IoD extends security while preserving computation and communication efficiencies.

Keywords: IoD, UAV, Key Establishment, Device Access Control, Stolen IoT device

1. Introduction

The notion of the internet of things (IoT) is that everything around us is accessible and connected with a global network. We are ever more surrounded by numerous tiny objects embedded with internet connectivity and intelligence, fostering convenience to human beings in everyday lives. IoT has influenced nearly every aspect of life, including, but not limited to, healthcare, smart city-based energy savings, farming, transportation, environment, search and rescue, surveillance and security monitoring, and domestic, business, and industrial economies. The objects in IoT ecosystem may be broadly categorized into physical objects and virtual objects. The physical objects may include drones, sensors, cameras, mobile devices, vehicles, etc. On the other hand, the virtual objects comprise agenda, e-ticket, e-wallet [1]. If adequately integrated with web applications, the virtual objects may assist the physical objects in communicating with one another and conferring with intelligence so that the physical objects may operate without human involvement.

An Unmanned Aerial Vehicle (UAV), commonly known as a drone, is a miniature aircraft without a human pilot. These drones are remotely controlled from the control room and are assigned a specific flying zone to report the control room information. An advanced drone or UAV is equipped with IoT-based sensors and actuators, computing and communication modules, micro-controllers, wireless transceivers, battery unit, inertial measurement unit (IMU) rotors recorder [2, 3]. The IoT-based sensors may comprise ultrasonic distance sensors, radio detection sensors, range ranging sensors, magnetic field detection sensors, temperature sensors, orientation sensors, and chemical sensors [4]. Recently, we witnessed some emerging use cases of drones in IoT realm, termed as "IoT-enabled drones" or "internet of drones," certifying that IoT is one of the enabling technologies for internet of drones (IoD). These IoT-enabled drones have been finding new applications ranging from fun toys, cinematography, sports-based photography, or multimedia entertainment to mission critical jobs-military missions, research, medical applications, and rescue operations. Some big companies such as Amazon and Google are increasingly playing their role in developing and refining drone delivery services. i.e., by initiating the projects such as Amazon Prime Air and the Project Wing, respectively. AT&T also employed drones to automate its cellular tower inspections. Recently, Dubai (UAE) introduced drones in the transportation arena by launching its flying taxi service [2]. Furthermore, research predicts that in 2025, the UAVs-based transactions' market size may go as high as 75-billion Yuan in China [5]. Nevertheless, the integration of IoT and UAVs might pose some security challenges. The attacks on the communication channels or authentication protocols for identifying drones may prove fatal. The security of IoT-enabled drones is crucial for the accomplishment of mission-critical,

*Corresponding author
Email address: yousafbinzikria@ynu.ac.kr (Yousaf bin Zikria)

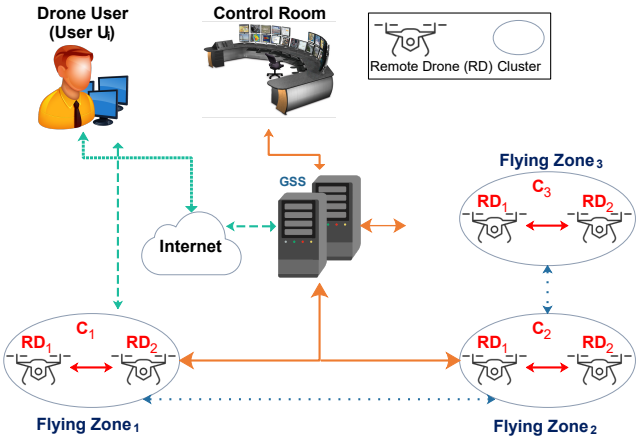

Figure 1: IoD Communication Architecture

safety-critical, or surveillance operations. The drones need to be tracked precisely for multiple reasons, such as avoiding collisions, identifying unauthorized drone flights, or increasing traffic efficiency. An attacker may disrupt the drone surveillance operation by identifying the drone location and disturbing the tracking services and or even attempt capturing it; for example, the Iranian military captured the U.S Lockheed Martin RQ-170 Sentinel drone [6]. In the discussed system architecture, the security and privacy issues among drones lying in respective flying zones, between drone and ground station server (GSS), and between GSS and control room (CR) in IoD environment need to be resolved.

2. Related work

In 2019, Srinivas et al.[7] proposed a temporal-credential (TC) and symmetric key based authentication framework for Internet of Drones (IoD); whereas, same was proved helpless against impersonation, once the verifier is stolen [8]. Moreover, the proposal of Srinivas et al. [7] was also debated for lack of untraceability by Ali et al. [8]. In the same year, Zhou et al. [9] also proposed an access control mechanism between a user and a node in distributed IoT settings through bilinear pairing. The weaknesses of Zhou et al.'s scheme against responder/IoT node impersonation was argued in [10]. Similarly, in 2019 another IoD access control mechanism was devised by Wazid et al. [11]. In a very similar manner to Srinivas et al., the scheme of Wazid et al. is insecure against impersonation launched after a successful stolen verifier attack. In 2020, Zhang et al. [12] also proposed another IoD authentication scheme using only symmetric key primitives. The scheme of Zhang et al. lacks perfect forward secrecy. Zhang et al.'s scheme are also weak against stolen verifier and insider attacks. Bera et al. [13] also proposed a scheme to secure IoD environment using certificate based access control and blockchains. Due to the usage of static pseudo identity (RID_{DRi}), the scheme of Bera et al. cannot provide user/drone anonymity. Some other schemes [14–19] were also designed for authentication/access control in IoT based systems. The scheme of Challa et al. [14] was proposed using ECC based signatures to provide access control among two entities. However, Chaudhry et al. [19] argued the inapplicability of the scheme of Challa et al. for IoT based systems due to a critical flaw in their scheme. In 2020, another scheme to provide authentication in IoD scenario was proposed by Tanveer et al.[20]. The scheme of Tanveer et al. was built over symmetric key functions. Due to usage of static identity (SID_{MS}) of the Management Server (MS) and publicly shared timestamp, the original identity SID_{MUi} of the mobile user/device can be exposed on the fly, which leads to non-provision of anonymity by the scheme of Tanveer et al. Recently, in this context, Bera et al. [21] also presented another access control scheme for IoT-enabled IoD environment and managed the post authentication data through blockchain technology (BT). The scheme provides a good blend of BT, IoT, and IoD systems and provides good access control to the user for drones.

2.1. Motivation and Contributions

It is inevitable to deploy some new drone nodes on a dynamic basis in the IoD environment. These drones are susceptible to physical capture threats or undergo any hardware failure or power consumption outages. In flying zones, the deployed drones may not always be legitimate nodes since the malicious drones or nodes can also be deployed instead by active adversaries. Therefore, it would be quite hard to discern a genuine node in many malicious nodes in IoD networks. This calls for designing an effective access control mechanism regarding the placement of new drone nodes to prevent malicious node entry in the IoD environment.

Earlier, Lin et al. [3] identified different security problems in IoT-enabled drones system. The focus of [3] was to address critical security concerns such as data confidentiality, privacy maintenance, and flexible accessibility. Thus, any presented model must accommodate key management, access control, privacy concerns, and intrusion detection for IoD security. In this connection, Bera et al. [21] also presented an IoD access control method. However, it suffers many security limitations, including insecurities against drone impersonation, the man in the middle, and replay attacks along with non-provision of user anonymity, as proved in forthcoming sections. We then proposed an improved certificate based access control scheme to extend secure device/drone to device/drone (D2D) and drone to GSS sessions. The proposed scheme provides authentication and a key agreement between the entities on the Internet of Drones (IoD) environments. Following are the contributions of this study:

1. We reviewed and found some critical security flaws in Bera et al.'s scheme [21].
2. We proposed GCACS-IoD, an enhanced and secure access control scheme for IoT-enabled drones environment, by employing the certificates issued by the control room (CR). Our scheme assures the mutual authenticity and key agreement among drones and between drones and corresponding GSS.

Table 1: Notation guide

Notations	Description
$E_q(a,b),\ G$	Elliptic Curve, Base point over $E_q(a,b)$
CR, ID_{CR}, FZ_i	Control Room, identity of CR, Flying Zone
GSS, ID_{GSS}	Ground Station Server, identity of GSS
DR_k, ID_{DRk}	K^{th} Drone, identity of DR_k
r_{CR}, Pub_{CR}	private, public key pair of CR
r_{DRk}, Pub_{DRk}	private, public key pair of DR_k
r_{GSS}, Pub_{GSS}	private, public key pair of GSS
TC_{DRk}, TC_{GSS}	Temporal credentials of DR_k and GSS
$Cert_{DRk}, Cert_{GSS}$	Certificates of DR_k and GSS
RTS_{DRk}, RTS_{GSS}	Registration timestamps of DR_k and GSS
MK_x, E_{Pubx}	Mater key of x, Encryption/Decryption
$TS_x,\ \Delta T$	Timestamp at entity x, Delay tolerance
$\|\|, h(.), \oplus$	concat, hash, xor operators
$\mathcal{A}, \stackrel{?}{=}$	Adversary, Equality Check

3. We employ the Real-or-Random (ROR) based random oracle model to formally evaluate the protocol and security of mutually agreed session key (SK).

4. We also present the informal discussion on the protocol's security features concerning immunity to threats, such as forgery attacks, privileged insider attacks, man-in-the-middle (MIDM) attacks, physical drone capture attacks, replay attacks, and session specific temporary information threat.

5. In the end, we present a comparative evaluation and analysis of computational and communicative costs as well as security-based functionality attributes of previous schemes with the proposed scheme.

2.2. System Model

In IoD architecture, as shown in Fig. 1, before the deployment, the drones and the Ground station servers (GSSs) are registered with Control Room (CR)- a trusted authority. The network structure as adopted from Garibi et al. [22] consists of several flying zones, each managed by its' respective GSS. Typically, drones are deployed in some specified flying zone. After registration, the drones and it's respective GSS authenticate one another on a public wireless channel, which may raise some privacy concerns and access control issues in the IoD system. The users/decision makers can get surveillance or otherwise data from the GSS. It's the same data that GSS collects from several drones after performing a cycle of authentication with each drone. Similarly, two or more drones can communicate with each other to carry out a collective task like rescue services, etc. In this case, the drones can authenticate each other and share their data to perform collective tasks.

2.3. Attack Model

The simulated attack model in this paper is considered as per assumptions mentioned in [23–29] and described as follows:

1. The attacker $\mathcal{A}$ has completely controlled the public channel, and $\mathcal{A}$ is assumed to be powerful enough to read, modify, add new, and can replay an old transmitted message.

2. $\mathcal{A}$ by using power analysis can read the contents stored in physically captured drone [30, 31].

3. $\mathcal{A}$ knows all the public system parameters, including the identities of the drones, GSS and CR.

3. Review of the Scheme of Bera et al.

The following subsections are reserved to briefly explain different phases of the scheme of Bera et al. [21]. The processes involve three entities, namely: 1) Control Room (CR) is taken as the trusted authority responsible for initialization and provision of certificates to the communicating entities, 2) Ground Station Server (GSS) acts as the intermediate authority to provide user access to a drone, and 3) Drones flying in same or different flying zones which are responsible for performing application specific tasks like, monitoring, rescue services, etc. Before moving further, Table 1 may be consulted for the notations used in this paper.

3.1. Initialization Phase

For initialization of the system, the CR selects curve $E_q(a,b)$ along with a base point G. The CR then selects a secure hash operator $h(.)$, it's own private key r_{CR} and computes public key $Pub_{CR} = r_{CR}.G$. Finally, all public parameters are announced and CR keeps its private key confidential. CR also selects an identity ID_{CR}.

3.2. Registration phase

The CR registers all drones and corresponding GSS. Following subsections describe both the registration phases:

3.2.1. Drone registration

For each drone $\alpha : \{\alpha = 1, 2, 3...m\}$, the CR selects a corresponding master key MK_α, identity ID_α, private key $r_{DR\alpha} \in Z_q^*$ and computes public key $Pub_{DR\alpha} = r_{DR\alpha}.G$. CR further computes $TC_\alpha = h(MK_\alpha||ID_\alpha||ID_{GSS}||RTS_\alpha)$ and private key based certificate $Cert_\alpha = r_{DR\alpha} + h(Pub_{DR\alpha}||ID_{GSS}||Pub_{CR}) * r_{CR}$. The CR then selects t-degree bivariate polynomial $f_i(x,y) = \sum_{u=0}^{t}\sum_{v=0}^{t} a_{u,v}x^u y^b \mod q$, where $t >> m$ and computes $f_i(ID_\alpha, y)$ for each drone DR_α. Finally, CR stores $\{ID_\alpha, ID_{GSS}, Pub_{DR\alpha}, Pub_{GSS}, TC_\alpha, Cert_\alpha, f_i(ID_\alpha, y)\}$ in DR_α's memory. Now each DR_α is ready for deployment.

3.2.2. GSS registration

To register a GSS, the CR selects private key $r_{GSS} \in Z_q^*$ of GSS and computes public key $Pub_{GSS} = r_{GSS}.G$. CR further computes $TC_{GSS} = h(ID_{GSS}||ID_{CR}||r_{GSS}||RTS_{GSS})$ and private key based certificate $Cert_{GSS} = r_{GSS} + h(Pub_{GSS}||ID_{GSS}||Pub_{CR}) * r_{CR}$. The CR then computes $f_i(ID_{GSS}, y)$ for each GSS. Finally, CR stores $\{(ID_\alpha, Pub_{DR\alpha} : \forall\alpha = 1, 2, 3...m), ID_{GSS}, Pub_{GSS}, Pub_{CR}, TC_{GSS}, Cert_{GSS}, f_i(ID_{GSS}, y) : \forall i = 1, 2, 3..m\}$ in GSS's memory.

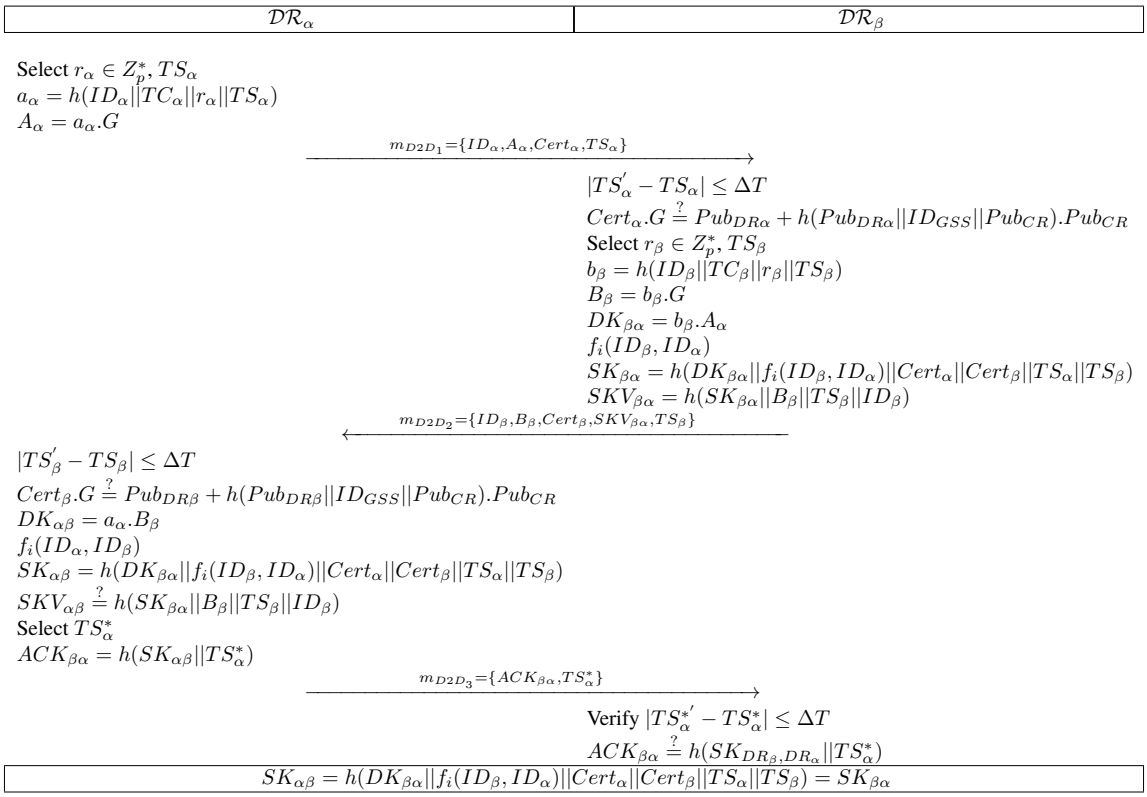

Figure 2: Bera et al.'s Procedure

3.3. D2D Access control phase

A drone say DR_α can initiate this phase to complete mutual authentication and key agreement with another neighboring drone say DR_β. The access control phase is shown in Fig. 2, as well as described below:

BDA 1: $\boldsymbol{DR_\alpha \rightarrow DR_\beta : \{m_{D2D_1}\}}$
DR_α selects random variable and timestamp pair $\{r_\alpha \in Z_p^*, TS_\alpha\}$ and computes $a_\alpha = h(ID_\alpha||TC_\alpha||r_\alpha||TS_\alpha)$, $A_\alpha = a_\alpha.G$. DR_α now sends $m_{D2D_1} = \{ID_\alpha, A_\alpha, Cert_\alpha, TS_\alpha\}$ to DR_β.

BDA 2: $\boldsymbol{DR_\beta \rightarrow DR_\alpha : \{m_{D2D_2}\}}$
On receiving $\{m_{D2D_1}\}$, DR_β checks time-freshness as $|TS'_\alpha - TS_\alpha| \leq \Delta T$, on failure aborts the session and on success checks validity of certificate as $Cert_\alpha.G \stackrel{?}{=} Pub_{DR\alpha}+h(Pub_{DR\alpha}||ID_{GSS}||Pub_{CR}).Pub_{CR}$. If certificate legality is proved, DR_β selects random variable and timestamp pair $\{r_\beta \in Z_p^*, TS_\beta\}$ and computes $b_\beta = h(ID_\beta||TC_\beta||r_\beta||TS_\beta)$, $B_\beta = b_\beta.G$, $DK_{\beta\alpha} = b_\beta.A_\alpha$ and $f_i(ID_\beta, ID_\alpha)$. DR_β now computes session key $SK_{\beta\alpha} = h(DK_{\beta\alpha}||f_i(ID_\beta, ID_\alpha)||Cert_\alpha||Cert_\beta||TS_\alpha||TS_\beta)$ and verifier of $SK_{\beta\alpha}$ as $SKV_{\beta\alpha} = h(SK_{\beta\alpha}||B_\beta||TS_\beta||ID_\beta)$. This step finishes normally after DR_β sends reply message $m_{D2D_2} = \{ID_\beta, B_\beta, Cert_\beta, SKV_{\beta\alpha}, TS_\beta\}$ to DR_α.

BDA 3: $\boldsymbol{DR_\alpha \rightarrow DR_\beta : \{m_{D2D_3}\}}$
On receiving $\{m_{D2D_2}\}$, DR_α checks time-freshness as $|TS'_\beta - TS_\beta| \leq \Delta T$, on failure aborts the session and on success checks validity of certificate as $Cert_{DR\beta}.G \stackrel{?}{=} Pub_{DR\beta}+h(Pub_{DR\beta}||ID_{GSS}||Pub_{CR}).Pub_{CR}$. If certificate legality is proved, DR_α computes $DK_{\alpha\beta} = a_\alpha.B_\beta$ and $f_i(ID_\alpha, ID_\beta)$. DR_α now computes session key $SK_{\alpha\beta} = h(DK_{\beta\alpha}||f_i(ID_\beta, ID_\alpha)||Cert_\alpha||Cert_\beta||TS_\alpha||TS_\beta)$ and verifies its' validity through checking $SKV_{\alpha\beta} \stackrel{?}{=} h(SK_{\beta\alpha}||B_\beta||TS_\beta||ID_\beta)$ and on success, DR_α generates TS^*_α and computes the verifier $ACK_{\beta\alpha} = h(SK_{\alpha\beta}||TS^*_\alpha)$ and sends $m_{D2D_3} = \{ACK_{\beta\alpha}, TS^*_\alpha\}$ to DR_β.

BDA 4: On receiving $\{m_{D2D_3}\}$, DR_β checks time-freshness as $|TS^{*'}_\alpha - TS^*_\alpha| \leq \Delta T$, on failure aborts the session and on success checks validity of verifier $ACK_{\beta\alpha} \stackrel{?}{=} h(SK_{\beta\alpha}||TS^*_\alpha)$, if it holds DR_β authenticates the legality of DR_α and keeps $SK_{\alpha\beta}$ as the shared key with DR_α for secure communication.

Remark: The drone to the ground station server (GSS) access phase in Bera et al.'s scheme is very similar to their drone to drone access control phase, and in the drone to GSS access phase, both parties verify each others' certificate. Therefore, this phase is not being described in this article. However, interested readers may consult the original article by Bera et al.

4. Weaknesses of Bera et al.'s scheme

This section proves some of the security weaknesses of the scheme of Bera et al. to show that their scheme is defense-

less against several attacks including: 1)drone/GSS impersonation, 2)man in the middle, and 3)replay attacks. Moreover, their scheme lacks user anonymity.

4.1. Drone Impersonation attack

This subsection shows that in Bera et al.'s scheme, an active attacker just by listening to the communication channel can successfully impersonate on behalf of any drone and exchange session keys with counterparts. The attack simulation is as follows:

BIA 1: Let some drone say DR_α initiates an authentication request by transmitting $m_{D2D_1} = \{ID_\alpha, A_\alpha, Cert_\alpha, TS_\alpha\}$ to DR_β, which after processing the request, responds with $m_{D2D_2} = \{ID_\beta, B_\beta, Cert_\beta, SKV_{\beta\alpha}, TS_\beta\}$. Let attacker $\mathcal{A}$ listens the public channel and records both request and response messages. Now $\mathcal{A}$ can impersonate any of the drone by just using it's identity and certificate.

BIA 2: To impersonate on behalf of DR_α, $\mathcal{A}$ selects fresh $TS_{\overline{\alpha}}$, generates $a_{\overline{\alpha}}$ and computes $A_{\overline{\alpha}} = a_{\overline{\alpha}}.G$. Then $\mathcal{A}$ sends $m_{\overline{D2D}_1} = \{ID_\alpha, A_{\overline{\alpha}}, Cert_\alpha, TS_{\overline{\alpha}}\}$ to DR_β.

BIA 3: On receiving $\{m_{\overline{D2D}_1}\}$, DR_β checks time-freshness $|TS'_{\overline{\alpha}} - TS_{\overline{\alpha}}| \leq \Delta T$, as the $TS_{\overline{\alpha}}$ is freshly generated, so $\mathcal{A}$ passes this test. Now, DR_β checks the validity of certificate $Cert_\alpha.G \stackrel{?}{=} Pub_{DR\alpha} + h(Pub_{DR\alpha}||ID_{GSS}||Pub_{CR}).Pub_{CR}$, as certificate is genuine so $\mathcal{A}$ passes this test too. DR_β now computes and send $m_{D2D_2} = \{ID_\beta, B_\beta, Cert_\beta, SKV_{\beta\alpha}, TS_\beta\}$ to DR_α

BIA 4: $\mathcal{A}$ intercepts the message and computes $DK_{\alpha\beta} = a_{\overline{\alpha}}.B_\beta$ and $f_i(ID_\alpha, ID_\beta)$. $\mathcal{A}$ now computes session key $SK_{\alpha\beta} = h(DK_{\alpha\beta}||f_i(ID_\beta, ID_\alpha)||Cert_\alpha||Cert_\beta||TS_{\overline{\alpha}}||TS_\beta)$, $ACK_{\beta\alpha} = h(SK_{\alpha\beta}||TS^*_\alpha)$ and sends $m_{\overline{D2D}_3} = \{ACK_{\beta\alpha}, TS^*_{\overline{\alpha}}\}$ to DR_β.

BIA 5: On receiving $\{m_{\overline{D2D}_3}\}$, DR_β checks time-freshness as $|TS^{'*}_\alpha - TS^*_{\overline{\alpha}}| \leq \Delta T$, as the $TS^*_{\overline{\alpha}}$ is freshly generated, so $\mathcal{A}$ passes this test. Finally. DR_β checks validity of verifier $ACK_{\beta\alpha} \stackrel{?}{=} h(SK_{\beta\alpha}||TS^*_\alpha)$. It is obvious that $\mathcal{A}$ knows all involved parameters. So, will pass this test and DR_β authenticates the legality of DR_α and keeps $SK_{\alpha\beta}$ as the shared key with DR_α for secure communication.

Therefore, $\mathcal{A}$ has successfully impersonated on behalf of DR_α. In the very similar way, $\mathcal{A}$ can impersonate on behalf of DR_β and/or GSS.

4.2. Man in middle attack

For launching man in middle attack, the attacker can just intercept the sender and receiver messages and, on both sides, send the modified parameter A_α and B_β. The attacker can establish two connections, one with each of the drones.

4.3. Replay attack

In Bera et al.'s scheme, the drone sends $m_{D2D_1} = \{ID_\alpha, A_\alpha, Cert_\alpha, TS_\alpha\}$ and the receiving drone/GSS just verifies the timestamp and certificates. After intercepting m_{D2D_1}, an attacker can replay it by replacing the old timestamp with the fresh one. Once received, the receiver (drone/GSS) verifies the time stamp. As It is freshly generated, it may check the certificate, which is also legal and may easily pass the verification test. Therefore, the receiver may process this replay message with an updated timestamp.

4.4. Lack of Anonymity

In Bera et al.'s scheme, the drone sends its' identity in plain text over an insecure channel. Hence, it does not provide drone anonymity as well as untraceability.

5. Proposed GCACS-IoD

This section presents GCACS-IoD, a generic certificate based inter-drone and drone to GSS authentication scheme proposed in this paper. The scheme can provide a secure session among any two entities (drone-drone, drone-GSS, GSS-drone) in the IoD environment. For simplicity, we keep the notations $\mathcal{DR}_\alpha$ and $\mathcal{DR}_\beta$ to show the working of the scheme between two drones, i.e., $\mathcal{DR}_\alpha$ and $\mathcal{DR}_\beta$, whereas anyone of the initiator or responder can be replaced by GSS. The details of the GCACS-IoD is as follows:

5.1. Registration phase

The CR registers all drones and corresponding GSS. Following subsections describe both the registration phases:

5.1.1. Drone registration

For each drone $\alpha : \{\alpha = 1, 2, 3...m\}$, the CR selects a corresponding master key MK_α, identity ID_α, private key $r_\alpha \in Z^*_q$ and computes public key $Pub_{DR\alpha} = r_\alpha.G$. CR further computes private key based certificate $Cert_\alpha = r_{CR} + h(Pub_{DR\alpha}||ID_{GSS}||ID_\alpha||Pub_{CR}) * r_{CR}$. Finally, CR stores $\{ID_\alpha, ID_{GSS}, r_\alpha, Pub_{DR\alpha}, Pub_{GSS}, Cert_\alpha\}$ in DR_α's memory. Now each DR_α is ready for deployment.

5.1.2. GSS registration

To register a GSS, the CR selects private key $r_{GSS} \in Z^*_q$ of GSS and computes public key $Pub_{GSS} = r_{GSS}.G$. CR further computes private key based certificate $Cert_{GSS} = r_{GSS} + h(Pub_{GSS}||ID_{GSS}||Pub_{CR}) * r_{CR}$. Finally, CR stores $\{(ID_{DR_\alpha}, Pub_{DR_\alpha} : \forall j = 1, 2, ...m), ID_{GSS}, r_{GSS}, Pub_{GSS}, Pub_{CR}, Cert_{GSS}\}$ in GSS's memory.

5.2. D2D Access control

A drone says DR_α can initiate this phase to complete mutual authentication and key agreement with another neighboring drone, say DR_β. The access control phase is shown in Fig. 3, as well as described below:

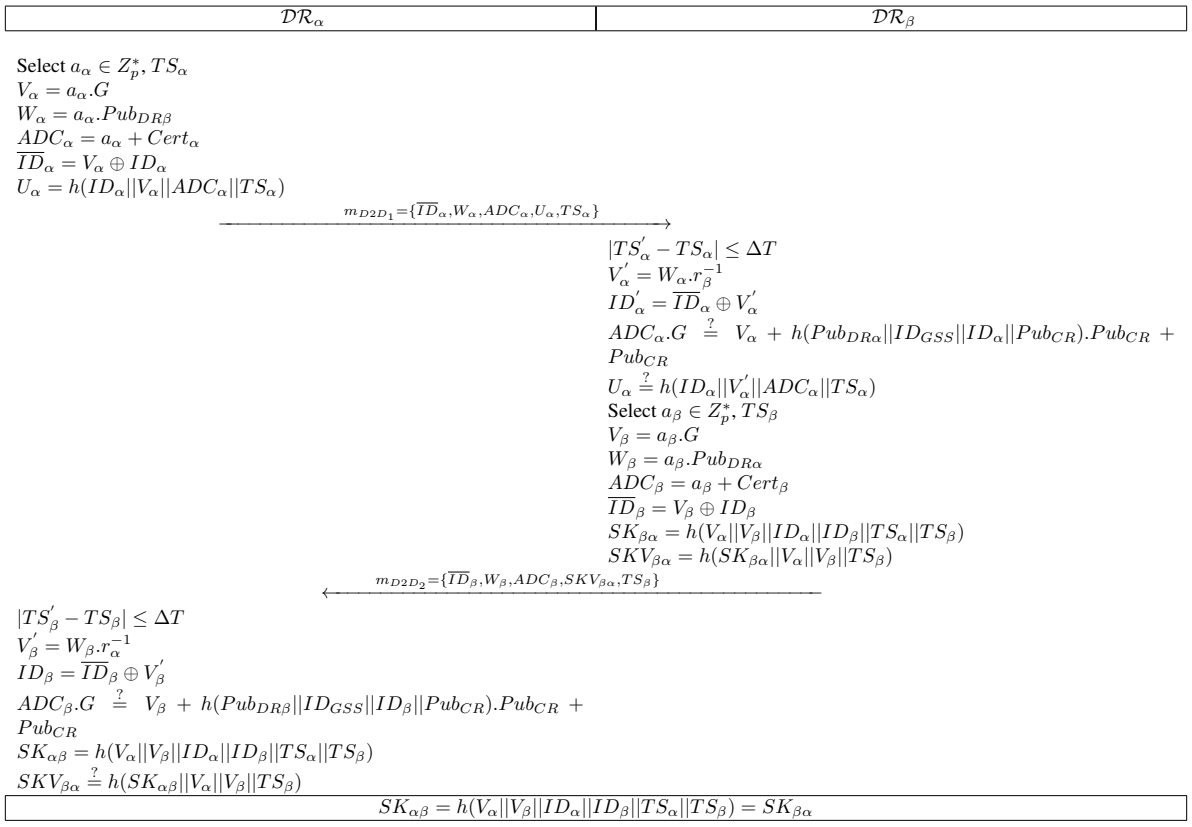

Figure 3: Proposed Procedure

PAC 1: $\boldsymbol{DR_\alpha \rightarrow DR_\beta : \{m_{D2D_1}\}}$
DR_α selects random variable and timestamp pair $\{a_\alpha \in Z_p^*, TS_\alpha\}$ and computes $V_\alpha = a_\alpha.G, W_\alpha = a_\alpha.Pub_{DR\beta}$, dynamic certificate $ADC_\alpha = a_\alpha + Cert_\alpha$, $\overline{ID}_\alpha = V_\alpha \oplus ID_\alpha$ and $U_\alpha = h(ID_\alpha||V'_\alpha||ADC_\alpha||TS_\alpha)$. DR_α now sends $m_{D2D_1} = \{\overline{ID}_\alpha, W_\alpha, ADC_\alpha, U_\alpha, TS_\alpha\}$ to DR_β.

PAC 2: $\boldsymbol{DR_\beta \rightarrow DR_\alpha : \{m_{D2D_2}\}}$
On receiving $\{m_{D2D_1}\}$, DR_β checks time-freshness as $|TS'_\alpha - TS_\alpha| \leq \Delta T$, on failure aborts the session and on success computes $V'_\alpha = W_\alpha.r_\beta^{-1}$, $ID_\alpha = \overline{ID}_\alpha \oplus V'_\alpha$ and checks validity of dynamic certificate as $ADC_\alpha.G \stackrel{?}{=} V_\alpha + h(Pub_{DR\alpha}||ID_{GSS}||ID_\alpha||Pub_{CR}).Pub_{CR} + Pub_{CR}$. If certificate legality is proved, DR_β further checks $U_\alpha \stackrel{?}{=} h(ID_\alpha||V'_\alpha||ADC_\alpha||TS_\alpha)$ and on success, DR_β selects random variable and timestamp pair $\{a_\beta \in Z_p^*, TS_\beta\}$ and computes $V_\beta = a_\beta.G, W_\beta = a_\beta.Pub_{DR\alpha}$, $ADC_\beta = a_\beta + Cert_\beta$ and $\overline{ID}_\beta = V_\beta \oplus ID_\beta$. DR_β now computes session key $SK_{\beta\alpha} = h(V_\alpha||V_\beta||ID_\alpha||ID_\beta||TS_\alpha||TS_\beta)$ and verifier of $SK_{\beta\alpha}$ as $SKV_{\beta\alpha} = h(SK_{\beta\alpha}||V_\alpha||V_\beta||TS_\beta)$. This step finishes normally after DR_β sends the reply message $m_{D2D_2} = \{\overline{ID}_\beta, W_\beta, ADC_\beta, SKV_{\beta\alpha}, TS_\beta\}$ to DR_α.

PAC 3: On receiving $\{m_{D2D_2}\}$, DR_α checks time-freshness as $|TS'_\beta - TS_\beta| \leq \Delta T$, on failure aborts the session and on success computes $V'_\beta = W_\beta.r_\alpha^{-1}$, $ID_\beta = \overline{ID}_\beta \oplus V'_\beta$ and checks validity of certificate as $ADC_\beta.G \stackrel{?}{=} V_\beta + h(Pub_{DR\beta}||ID_{GSS}||ID_\beta||Pub_{CR}).Pub_{CR} + Pub_{CR}$. If certificate legality is proved, DR_α computes session key $SK_{\alpha\beta} = h(V_\alpha||V_\beta||ID_\alpha||ID_\beta||TS_\alpha||TS_\beta)$ nd verifies its' validity through checking $SKV_{\beta\alpha} \stackrel{?}{=} h(SK_{\alpha\beta}||V_\alpha||V_\beta||TS_\beta)$, if it holds DR_α authenticates the legality of DR_β and keeps $SK_{\alpha\beta}$ as the shared key with DR_β for secure communication.

6. Security Analysis

This section proves the security of the proposed GCACS-IoD using a formal method. Moreover, a discussion on attack resilience and security features of the proposed scheme is also provided in the following subsections:

6.1. Formal Security Analysis

The commonly accepted Real-Or-Random ROR oracle model [32] as adopted in [33, 19, 28] is used to show that the proposed protocol is secure to extract the session key $SK_{\alpha\beta}$ between a drone DR_k and a ground station server GSS as well as the session key $SK_{\alpha\beta}$ between $\mathcal{DR}_\alpha$ and $\mathcal{DR}_\beta$ drones against an attacker $\mathcal{A}$. To achieve this goal, we investigate the ROR model first using the semantic security approach and then the session key security of the proposed protocol in Theorem 1. All queries described below will be executed by adversary $\mathcal{A}$. Furthermore,

as stated in [34], access to a collision resistant one way cryptographic hash function $h(.)$ is also given to all participants, including the adversary $\mathcal{A}$ and hash function $h(.)$ can be modeled as a RO, say $HASH$. The ROR model is performed using the following elements described as:

Participants. The entities, namely as drones DR_k and the GSS, are engaged at the time of the login and authentication process apart from the CR, which is involved only during the registration stage and the dynamic drones addition phases. $\Pi_{DR}^{b_1}$ and $\Pi_{GSS}^{b_2}$ are used to show the b_1 and b_2 instances of DR and GSS, commonly. These are called random oracles instances.

Execute$(\Pi_{DR}^{b_1}, \Pi_{GSS}^{b_2})$ The attacker can eavesdrop the messages shared between the DR and GSS by applying this query.
CorruptDrone$(\Pi_{DR}^{b_1})$ The attacker will steal secret parameters stores in the memory of a compromised or lost DR by applying this query. **Reveal**(Π^b) Attacker can be revealed a session-key $\mathcal{SK}_{DR_k,GSS}$ between GSS and DR_k or the session-key $SK_{\mathcal{DR}_\alpha,\mathcal{DR}_\beta}$ between drones $\mathcal{DR}_\alpha$ and $\mathcal{DR}_\beta$ shared between Π^b and its respective participant by applying this query.
Test(Π^b) By applying this query, the attacker $\mathcal{A}$ is allowed to call Π^b to test the originality of a session key and Π^b will have a random outcome of a flipped impartial coin, say d. **Accepted State**. If the last message of the valid protocol is accepted, the instance Π^b goes to its "'accepted state". When all the messages sent and received can be organized sequentially, they form the session identification Sid of Π^b for the currently executed session together.
Partnering. If the following three properties are true, the instances Π^{b_1} and Π^{b_2} are said to be participants with each other:
-Π^{b_1} and Π^{b_2} are to be in accepted states.
-Π^{b_1} and Π^{b_2} are to share the same Sid.
-Π^{b_1} and Π^{b_2} are mutual participants of each other.
Freshness. An instances Pi_{DR}^{b1} or Pi_{DR}^{b2} is fresh, if the attacker $\mathcal{A}$ can not determine a session key formed between two partnering entities using the $reveal(\Pi^b)$ query shown above.
The improved scheme's $(GCACS-IoD)$ semantic security is given in Definition 1 before proving Theorem 1.
Definition ($Semantic\ Security$). Let $ADV_{\mathcal{A}}^{GCACS-IoD}(l_p)$ be taken as $\mathcal{A}'s$ advantages who runs in polynomial time l_p for breaking the semantic-security of $GCACS-IoD$ in order to extract the session key between DR_k and GSS or session key between drones $\mathcal{DR}_\alpha$ and $\mathcal{DR}_\beta$ $ADV_{\mathcal{A}}^{GCACS-IoD}(l_p) = |2Pr[d^{'} = d] - 1|$. Here, d and $d^{'}$ represent the right and guessed bits, respectively.

Theorem 1. Let an attacker $\mathcal{A}$ tries to extract the session key $\mathcal{SK}_{DR_k,GSS}$ between DR_k and GSS or the session key $\mathcal{SK}_{\mathcal{DR}_\alpha,\mathcal{DR}_\beta}$ between $\mathcal{DR}_\alpha$ and $\mathcal{DR}_\beta$ in polynomial time l_p in the improved scheme $GCACS-IoD$.
$ADV_{\mathcal{A}}^{GCACS-IoD}(l_p) \leq \frac{q_{hash}^2}{HASH} + ADV_{\mathcal{A}}^{ECDDHP-IoD}(l_p)$.

Proof. The proof of this theorem is interpreted in a manner similar to that given in [21, 34, 18, 35, 19]. Three games say $Game_n^{\mathcal{A}}$ for the attacker $\mathcal{A}$ are needed, n=1,2,3. If the $Success_{Game_n}^{\mathcal{A}}$ shows an event that adversary $\mathcal{A}$ can be guessed bit d in the game $Game_n^{\mathcal{A}}$ correctly, adversary $\mathcal{A}$'s advantages is winning $Game_n^{\mathcal{A}}$ in proposed scheme can be then expressed as follows: $ADV_{\mathcal{A},Game_n}^{GCACS-IoD} = Pr[Success_{Game_n}^{\mathcal{A}}]$. Hence, we illustrated each game below in following manners.

Game$_1^{\mathcal{A}}$. This game helps attacker $\mathcal{A}$ to perform the real attack under the ROR paradigm against $GCACS-IoD$. The attacker $\mathcal{A}$ needs to select a random bit d before starting $Game_1^{\mathcal{A}}$. The semantic security specified in Definition 1 yields the results as set out below:

$$ADV_{\mathcal{A}}^{GCACS-IoD}(l_p) = |2ADV_{\mathcal{A},Game_1}^{GCACS-IoD}(l_p) - 1|. \quad (1)$$

Game$_2^{\mathcal{A}}$. This game $Game_2^{\mathcal{A}}$, an eavesdropping attack is simulated by attacker someone who can use Execute query to intercept all exchanged messages during the login and authentication process. The attackers $\mathcal{A}$ can eavesdrop all communication messages transmitted between $\mathcal{DR}_\alpha$ and $\mathcal{DR}_\beta$ drones: $m_{D2D_1} = \{\overline{ID}_\alpha, W_\alpha, ADC_\alpha, TS_\alpha\}$ and $m_{D2D_2} = \{\overline{ID}_\beta, W_\beta, ADC_\beta, SKV_{\beta\alpha}, TS_\beta\}$, try to build session key $SK_{\alpha\beta} = h(V_\alpha || V_\beta || ID_\alpha || ID_\beta || TS_\alpha || TS_\beta) = SK_{\beta\alpha}$. Then, with the aid of $reveal$ and $Test$ queries, the attacker must verify whether the extracted session key is a right one, or just a random key. Since all temporal and long-term secrets are covered by $h(.)$, even interception of m_{D2D_1} and m_{D2D_2} communications do not lead to raising the probability of success in the estimation of session keys $\mathcal{SK}_{DR_\alpha,DR_\beta}$. In the wake of the eavesdropping attack, $Game_1^{\mathcal{A}}$ and $Game_2^{\mathcal{A}}$ prove indistinguishable. That carries the following.

$$ADV_{\mathcal{A},Game_2}^{GCACS-IoD} = ADV_{\mathcal{A},Game_1}^{GCACS-IoD} \quad (2)$$

Game$_3^{\mathcal{A}}$. This game refers to an active attack in which we have $HASH$ and $CorruptDrone$ queries test simulations, and $ECDDHP$. To extract the session-key $\mathcal{SK}_{\alpha\beta}$, attacker needs to derive V_β and W_β. Let attacker $\mathcal{A}$ has already all the messages m_{D2D_1} and m_{D2D_2}. From intercepted messages to derive session key, attacker needs to solve the $ECDDHP$ in time l_p which has advantages probability $ADV_{\mathcal{A}}^{GCACS-IoD}(l_p)$. In the similar way, attacker $\mathcal{A}$'s chances in solving the $ECDDHP$ to extract V_β and W_β from the intercepted messages will be again $ADV_{\mathcal{A}}^{GCACS-IoD}(l_p)$. Consequently, these parameters are enclosed in $h(.)$. The hash values $h(.)$ are also unique, owing to timestamps and random numbers used in each message during a session's communications. Additionally, using the $CorruptDrone$ queries, attacker $\mathcal{A}$ will have the secret parameters that will be helpful in deriving the session keys, as random GSS secrets and temporary passwords as well as other non-corrupted drones are also important. When we exclude the simulation of $HASH$ and $CorruptDrone$ requests, it is worth remembering that both $Game_2^{\mathcal{A}}$ and $Game_3^{\mathcal{A}}$ are identical, so solving $ECDDHP$ is a simple task. The following relation is obtained using the results of the birthday paradox to find the hash collision and the benefit of solving $ECDDHP$:

$$ADV_{\mathcal{A},Game_2}^{GCACS-IoD} - ADV_{\mathcal{A},Game_3}^{GCACS-IoD} \leq \frac{q_{hash}^2}{2|HASH|} + ADV_{\mathcal{A}}^{ECDDHP-IoD}(l_p). \quad (3)$$

Since all the queries are already made by the attacker $\mathcal{A}$ and it only remains for the attacker $\mathcal{A}$ to guess a bit correctly for winning the $\mathbf{Game}_3^{\mathcal{A}}$. Therefore, it is obvious that

$$ADV_{\mathcal{A},Game_3}^{GCACS-IoD} = \frac{1}{2}. \quad (4)$$

Eq. (1)gives

$$\frac{1}{2}ADV_{\mathcal{A}}^{GCACS-IoD}(l_p) = |ADV_{\mathcal{A},Game_1}^{GCACS-IoD} - \frac{1}{2}|. \quad (5)$$

Eqs. (3)-(5) and the triangular inequality will lead to the following computations:

$$\begin{aligned} &\frac{1}{2}ADV_{\mathcal{A}}^{GCACS-IoD}(l_p) = \\ &|ADV_{\mathcal{A},Game_1}^{GCACS-IoD} - ADV_{\mathcal{A},Game_3}^{GCACS-IoD}| \\ &= |ADV_{\mathcal{A},Game_2}^{GCACS-IoD} - ADV_{\mathcal{A},Game_3}^{GCACS-IoD}| \\ &\leq \frac{q_{hash}^2}{2|HASH|} + ADV_{\mathcal{A}}^{ECDDHP-IoD}(l_p). \end{aligned} \quad (6)$$

Finally, multiplying the Eq. (6) by 2, the result is obtained as: $ADV_{\mathcal{A}}^{GCACS-IoD}(l_p) \leq \frac{q_{hash}^2}{|HASH|} + 2ADV_{\mathcal{A}}^{ECDDHP-IoD}(l_p)$.

6.2. Functional Security Provision

This subsection explains the security features and attack resilience of the proposed scheme.

6.2.1. Drone (Initiator) impersonation attack

In proposed GCACS-IoD, an adversary may attempt to construct an authorized authentication request $m_{D2D_1} = \{\overline{ID}_\alpha, W_\alpha, ADC_\alpha, U_\alpha, TS_\alpha\}$ to impersonate as a legal drone DR_α, where the parameters in m_{D2D_1} are computed as $W_\alpha = a_\alpha.Pub_\beta$, $ADC_\alpha = a_\alpha + Cert_\alpha$, $\overline{ID}_\alpha = V_\alpha \oplus ID_\alpha$, $U_\alpha = h(ID_\alpha||V_\alpha||ADC_\alpha||TS_\alpha)$. In order to compute this message, the adversary needs to access the certificate $Cert_\alpha$ and utilize it in the construction of m_{D2D_1}. However, by listening to the intercepted messages on an insecure channel, the adversary might not approach a valid certificate $Cert_\alpha$ to use it for malicious objectives. Hence, our scheme is immune to drone/initiator impersonation attacks.

6.2.2. Replay attack

In this scheme, the messages $m_{D2D_1} = \{\overline{ID}_\alpha, W_\alpha, ADC_\alpha, U_\alpha, TS_\alpha\}$ and $m_{D2D_2} = \{\overline{ID}_\beta, W_\beta, ADC_\beta, SKV_\beta, TS_\beta\}$ are exchanged between drones over an insecure channel in authentication phase. An adversary might intercept and replay these messages for impersonating legitimate drones by modifying a few parameters such as timestamps TS_α and TS_β. However, each of those messages (m_{D2D_1} and m_{D2D_2}) is designed by utilizing fresh timestamps and random nonces, which is duly verified by other corresponding drones.Beside the verification of timestamps in $U_\alpha \stackrel{?}{=} h(ID_\alpha||V_\alpha||ADC_\alpha||TS_\alpha)$ and $SKV_{\beta\alpha} \stackrel{?}{=} h(SK_{\alpha\beta}||V_\alpha||V_\beta||TS_\beta)$, the nonces a_α and a_β are also validated along with the verification of certificates $Cert_\alpha$ and $Cert_\beta$, i.e. $ADC_\alpha.G \stackrel{?}{=} V_\alpha + h(Pub_{DR\alpha}||ID_{GSS}||ID_\alpha||Pub_{CR}).Pub_{CR} + Pub_{CR}$ and $ADC_\beta.G \stackrel{?}{=} V_\beta + h(Pub_{DR\beta}||ID_{GSS}||ID_\beta||Pub_{CR}).Pub_{CR} + Pub_{CR}$, respectively. Hence, the adversary's attempt to replay the previously intercepted valid messages may be successfully thwarted on the receiver's end. Thus, our scheme is resilient against the replay attack.

6.2.3. Man-in-the-middle attack

In case, an adversary eavesdrops messages $m_{D2D_1} = \{\overline{ID}_\alpha, W_\alpha, ADC_\alpha, U_\alpha, TS_\alpha\}$ and $m_{D2D_2} = \{\overline{ID}_\beta, W_\beta, ADC_\beta, SKV_\beta, TS_\beta\}$, and attempts to modify or adapt its contents to deceive the legitimate participants, the former will not be able to accomplish its malevolent goals. This is because, if $\mathcal{A}$ attempts to append a fresh timestamp $TS_\alpha*$ ($TS_\beta*$) with the message m_{D2D_1} (m_{D2D_2}), or computes $\overline{ID}_\alpha*$ ($\overline{ID}_\beta*$) , $W_\alpha*$ ($W_\beta*$), $ADC_\alpha*$ ($ADC_\beta*$), and $U_\alpha*$ ($SKV_{\beta\alpha*}$) with fresh random integer $a_\alpha*$ ($a_\beta*$), it will be detected on another drone during the verification of $ADC_\alpha.G \stackrel{?}{=} V_\alpha + h(Pub_{DR\alpha}||ID_{GSS}||ID_\alpha||Pub_{CR}).Pub_{CR} + Pub_{CR}$ or by verifying $ADC_\beta.G \stackrel{?}{=} V_\beta + h(Pub_{DR\beta}||ID_{GSS}||ID_\beta||Pub_{CR}).Pub_{CR} + Pub_{CR}$. Any modification in the timestamp shall preclude the recovery of true identity ID_α (ID_β), that might invalidate the above comparison.

6.2.4. GSS (Responder) impersonation attack

An adversary may attempt to impersonate as a legal ground station server (GSS/ responder) by constructing the message $m_{D2D_2} = \{\overline{ID}_\beta, W_\beta, ADC_\beta, SKV_\beta, TS_\beta\}$. To construct this message, $\mathcal{A}$ may choose a random integer a_β and a fresh timestamp TS_β, and then compute $V_\beta = a_\beta.G$, $W_\beta = a_\beta.Pub_\alpha$. Nevertheless, to initiate a successful impersonation attack on authentication request of DR_α, $\mathcal{A}$ needs to get identity of the requesting drone and create dynamic certificate based on GSS's secret certificate. Computing ID_α from $\overline{ID}_\alpha = V_\alpha \oplus ID_\alpha$, the attacker needs private key of the GSS; whereas, to compute the dynamic certificate $ADC_\beta = a_\beta + Cert_\beta$, $\mathcal{A}$ needs secret certificate $Cert_\beta$ of the GSS. Hence, our protocol is resistant to GSS/responder impersonation attack.

6.2.5. Ephemeral secrets leakage attack

In proposed scheme, the session key between drones DR_α and DR_β is computed as $SK_{\alpha\beta} = h(V_\alpha||V_\beta||ID_\alpha||ID_\beta||TS_\alpha||TS_\beta)$, where $V_\alpha = a_\alpha.G$, $V_\beta = a_\beta.G$, ID_α and ID_β are identities, and TS_α and TS_β being the timestamps of DR_α and DR_β, respectively. In session key $SK_{\alpha\beta}$ ($SK_{\beta\alpha}$), the concatenated factor V_α (V_β) is computed from random nonce a_α (a_β) which is a short term secret of DR_α (DR_β) for the session, i.e. $V_\alpha = a_\alpha.G$ ($V_\beta = a_\beta.G$). Similarly, for the same session key $SK_{\alpha\beta}$ ($SK_{\beta\alpha}$), the concatenated factor V_β (V_α) is computed from r_α (r_β), a long term secret of DR_α (DR_β) in the protocol, i.e. $V_\beta = W_\beta.r_\alpha^{-1}$ ($V_\alpha = W_\alpha.r_\beta^{-1}$). It can

be witnessed that the session key is influenced by short term secrets as well as long term secrets in our scheme. Hence, the adversary will have to access both short term as well as long term secrets to compromise the session keys. Thus, our scheme is immune to ephemeral secrets leakage attacks.

6.2.6. Privileged insider attack

None of the drones submits its identity or registration parameters towards CR during the proposed scheme's registration phase. Rather, the CR initializes a drone, say DR_α, with precomputed credentials based on an assumed identity (ID_α) and private key (r_α) i.e. $\{\overline{ID}_\alpha, ID_\beta, r_\alpha, Pub_\alpha, Pub_\beta, Cert_\alpha\}$ prior to its deployment in network of drones. After this initialization procedure, all of the drones are deployed in IoD environment. In this manner, a malicious insider might not access any registration request parameters preloaded into the memory of drones before physical deployment. Hence, our scheme is naturally resistant to privileged insider attacks.

6.2.7. Mutual authentication

In our scheme, both participating drones DR_α and DR_β mutually establish an agreed session key $SK_{\alpha\beta} = SK_{\beta\alpha} = h(V_\alpha||V_\beta||ID_\alpha||ID_\beta||TS_\alpha||TS_\beta)$ after mutual authenticating each other. Before, finalizing the session key both entities authenticate one another on the basis of fresh timestamps (TS_α and TS_β) and random session nonces (a_α and a_β). The drone DR_β, after receiving m_{D2D_1} from DR_α checks the freshness of timestamp initially, and then computes $V_\alpha = W_\alpha.r_\beta^{-1}$ by taking inverse of W_α parameter using its private key r_β. Then, after deriving the DR_α's identity ID_α, the DR_β checks the equation $ADC_\alpha.G \stackrel{?}{=} V_\alpha + h(Pub_{DR\alpha}||ID_{GSS}||ID_\alpha||Pub_{CR})$. $Pub_{CR} + Pub_{CR}$ to verify the DR_α's authenticity. This not only verifies the certificate, but also the validity of short term session secret a_α as generated by DR_α. Likewise, The drone DR_α, after receiving m_{D2D_2} from DR_β checks the freshness of timestamp initially, and then computes $V_\beta = W_\beta.r_\alpha^{-1}$ by taking inverse of W_β parameter using its private key r_α. Then, after deriving the DR_β's identity ID_β, the DR_α checks the equation $ADC_\beta.G \stackrel{?}{=} V_\beta + h(Pub_{DR\beta}||ID_{GSS}||ID_\beta||Pub_{CR})$. $Pub_{CR}+Pub_{CR}$ to verify the DR_β's authenticity. This verifies not only the certificate but also the validity of short term session secret a_β as generated by DR_β. Thus, the mutual authenticity for both participating entities is ensured in our scheme.

6.2.8. Physical drone capture attack

As described in an attack model, an adversary A may physically attack the drone in a hostile environment and capture its stored contents. Thus, in our scheme, A may get the stored contents $\{ID_\alpha, ID_\beta, r_\alpha, Pub_\alpha, Pub_\beta, Cert_\alpha\}$ from the memory of compromised drone, say DR_α, by using a power analysis attack. Although the adversary gets access to those contents of compromised DR_α, it may not affect at all the communication or construction of session keys among other non-compromised drones in the IoD network system. Hence, our scheme is protected from physical drone capture attack.

6.2.9. Perfect forward secrecy

In case the private secret or long term secret of a drone is revealed to the adversary, the latter will not be able to compute the session key, which suggests that our scheme is compliant with perfect forward secrecy. This is because $\mathcal{A}$ requires to access both long term secrets "'r_α (r_β)"' as well as short term secrets "'a_α (a_β)"' for the compromised drone to recover a legitimate session key that was created by mutual authenticity with another drone. Alternatively, the adversary must compromise both drones' long-term secrets or private keys to construct the legal session key. Thus, the contributed scheme ensures the property of perfect forward secrecy.

6.2.10. Known key secrecy

In our scheme, if the adversary is able to compromise the current session key $SK_{\alpha\beta} = SK_{\beta\alpha} = h(V_\alpha||V_\beta||ID_\alpha||ID_\beta||TS_\alpha||TS_\beta)$, it may not compute the previous session keys as created before, between the same drones. To recover the previous session keys between those drones, A needs to compromise either the long term secrets of both drones, i.e., r_α and r_β, or all short term "'a_α (a_β)"' and long term secrets "'r_α (r_β)"' of a compromised drone. However, compromising many drones at the same time, or the simultaneous access to short and long term secrets of a compromised drone is a strong assumption and becomes infeasible for the adversary.

7. Comparative Performance and Security Analysis

This section is dedicated to showing the comparisons with respect to computation, communication, and security features extended by proposed and competing scheme [36–38, 35, 39, 21].

7.1. Computation Cost analysis

For the comparative computation cost analysis, following notations with their running time as per the experiment presented in [40] on a PC $E2200$ with Dual CPU, 2.20 GHz speed processor and 2 GB memory, performed on Ubuntu OS plus PBC Library:

- Multiplication on $E_q(a, b)$ point: $T_{mep} \approx 2.226\ ms$
- Addition on $E_\alpha(i, j)$ point$T_{aep} \approx 0.0288\ ms$
- Hash computation time$T_{hsh} \approx 0.0023\ ms$
- Bilinear pairing time: $T_{ebp} \approx 5.811\ ms$
- Exponentiation time $T_{exn} \approx 3.85\ ms$
- Enc/Decryption time: $T_{edn} \approx 0.0046\ ms$

Referring the experimental results presented in [40], the computation costs of proposed and competing schemes [36–38, 35, 39, 21] are given in Table 2. The proposed scheme completes the drone to the drone authentication process in approximately 17.9416 milliseconds (ms), which is just 0.053 ms higher than Bera et al.'s scheme.

Table 2: Computational Cost Analysis

Scheme	Drones	Total	RT	C_1	C_2
Huang et al.[36]	$4T_{mep} + 8T_{hsh}$	$4T_{mep} + 8T_{hsh}$	≈ 8.9224	4	1920
Li et al.[37]	$2T_{ebp} + 2T_{hsh}$	$6T_{ebp} + 3T_{mep} + 1T_{edn} + 2T_{hsh}$	≈ 41.6062	2	3488
Luo et al.[38]	$2T_{ebp} + 2T_{hsh}$	$4T_{ebp} + 3T_{mep} + 2T_{aep} + 2T_{hsh}$	≈ 29.9312	2	3040
Malani et al. [35]	$12T_{mep} + 4T_{aep} + 15T_{hsh}$	$12T_{mep} + 4T_{aep} + 15T_{hsh}$	≈ 26.8607	2	2144
Tian et al. [39]	$8T_{exn} + 9T_{hsh}$	$8T_{exn} + 9T_{hsh}$	≈ 30.8207	2	11712
Bera et al [21]	$8T_{mep} + 2T_{aep} + 10T_{hsh}$	$8T_{mep} + 2T_{aep} + 10T_{hsh}$	≈ 17.8886	3	1696
Proposed	$8T_{mep} + 4T_{aep} + 8T_{hsh}$	$8T_{mep} + 4T_{aep} + 8T_{hsh}$	≈ 17.9416	2	1664

Note: RT: Running time in ms; C_1: Number of message exchanges; C_2: Number of Bits exchanges.

7.2. Communication Cost

For communication cost purposes, the bits sent over communication media and the number of messages exchanged between the two parties are considered. The communication costs in Table 2 are accumulated by considering the size of identity, Hash function length, and random number as 160 bits. In contrast, the ECC point's size is taken as 320 bits long, and the timestamp length is fixed at 32 bits.Two message exchanges complete the authentication process for the proposed scheme: 1) the initiation message $m_{D2D_1} = \{\overline{ID}_\alpha, W_\alpha, ADC_\alpha, U_\alpha, TS_\alpha\}$ consumes $\{160+320+160+160+32\} = 832$ bits, and 2)the reply message $m_{D2D_2} = \{\overline{ID}_\beta, W_\beta, ADC_\beta, SKV_{\beta\alpha}, TS_\beta\}$ also needs same $\{160 + 320 + 160 + 160 + 32\} = 832$ bits to sent to the initiation drone. Therefore, the communication cost of the proposed scheme is 1664 bits with 2 message exchanges. Referring the Table 2 proposed scheme has lowest communication cost, when compared with related schemes [36–38, 35, 39, 21].

7.3. Security features comparison

The security features/requirements comparison of the proposed scheme with related schemes [36–38, 35, 39, 21] is furnished in Table 3. The illustration in Table 3 shows that the proposed scheme provides all security requirements, including direct device/drone to device/drone communication; whereas, all other schemes lack some of the security requirements. Like, the scheme proposed in [39] provides anonymity but lacks mutual authentication, temporary secret leakage attack as well lacks the formal proof of scheme security; whereas, the rest of the schemes [36–38, 35, 21] do not provide device/drone anonymity. Besides, the scheme [36] cannot resist the replay attack and deployment of the malicious device. The scheme proposed in [37] does not extend direct communication between the drones and lacks formal security proof and mutual authentication. The scheme proposed in [38] cannot resist the physical capturing of the device and temporary secrets leakage attacks as well as does not provide mutual authentication. The scheme proposed in [35] lacks resistance against the deployment of malicious device and drone/device impersonation attacks. The scheme of Bera et al. [21], as proved earlier in this paper, cannot extend resistance against replay, man in middle, and device/drone impersonation attacks and does not provide user anonymity.

8. Conclusion

This paper has briefly reviewed and cryptanalyzed a recent authentication scheme for securing the IoD environment. It has been proved in this paper that the scheme proposed by Bera et al. has many weaknesses, including insecurities against impersonation, the man in middle and replay attacks, as well as non provision of anonymity. A certificate based generic access control scheme usable in both drone to drone (D2D) and drone to GSS scenarios for IoD (GCACS-IoD) is then proposed. The proposed GCACS-IoD, while providing D2D direct communication, is free of any pairing operations. The security analysis of the proposed GCACS-IoD has been carried out using a formal RoR model along with a brief discussion on security features and attack resistance. The performance and security features comparisons of the proposed GCACS-IoD with related schemes showed that the proposed scheme resists known attacks and completes the access control process by exchanging only two messages. Consequently, it is best suitable for generic access control in IoD based systems.

References

[1] E. Bach, Toward a theory of pollard's rho method, Information and Computation 90 (2) (1991) 139–155.

[2] W. Hong, L. Jianhua, L. Chengzhe, W. Zhe, A provably secure aggregate authentication scheme for unmanned aerial vehicle cluster networks, Peer-to-Peer Networking and Applications 13 (1) (2020) 53–63.

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

[4] H. Z. Liao, Y. Y. Shen, On the elliptic curve digital signature algorithm, Tunghai Science 8 (2006) 109–126.

[5] Research (2016) china unmanned aerial vehicle industry research report (2016).
URL https://www.iresearch.com.cn/Detail/report?id=2588&isfree=0,2016

[6] K. Hartmann, C. Steup, The vulnerability of uavs to cyber attacks-an approach to the risk assessment, in: 2013 5th international conference on cyber conflict (CYCON 2013), IEEE, 2013.

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

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

[9] Y. Zhou, T. Liu, F. Tang, M. Tinashe, An unlinkable authentication scheme for distributed iot application, IEEE Access 7 (2019) 14757–14766.

Table 3: Security Comparisons

	[36]	[37]	[38]	[35]	[39]	[21]	Our
D2D Direct Communication	✓	✗	✓	✓	✓	✓	✓
Anonymity	✗	✗	✗	✗	✓	✗	✓
Mutual Authentication	✓	✗	✗	✓	✗	✓	✓
Resists Replay Attack	✗	✓	✓	✓	✓	✓	✓
Resists Device/Drone Physical Capture	✓	✓	✗	✓	✓	✓	✓
Resists Malicious Device	✗	✓	✓	✗	✓	✓	✓
Resists Man in middle attack	✓	✓	✓	✓	✓	✗	✓
Resists Device/Drone Impersonation	✓	✓	✓	✗	✓	✗	✓
Resists Temporary secrets leakage	✓	✓	✗	✓	✗	✓	✓
Extends Key Agreement	✓	✓	✓	✓	✓	✓	✓
Formal Security	✓	✗	✗	✓	✗	✓	✓

[10] S. A. Chaudhry, M. S. Farash, N. Kumar, M. H. Alsharif, Pflua-diot: A pairing free lightweight and unlinkable user access control scheme for distributed iot environments, IEEE Systems Journal (2020) 1–8doi:10.1109/JSYST.2020.3036425.

[11] M. Wazid, A. K. Das, N. Kumar, A. V. Vasilakos, J. J. P. C. Rodrigues, Design and analysis of secure lightweight remote user authentication and key agreement scheme in internet of drones deployment, IEEE Internet Things J. 6 (2) (2019) 3572–3584.

[12] Y. Zhang, D. He, L. Li, B. Chen, A lightweight authentication and key agreement scheme for internet of drones, Computer Communications (2020).

[13] B. Bera, S. Saha, A. K. Das, N. Kumar, P. Lorenz, M. Alazab, Blockchain-envisioned secure data delivery and collection scheme for 5g-based iot-enabled internet of drones environment, IEEE Transactions on Vehicular Technology 69 (8) (2020) 9097–9111.

[14] S. Challa, M. Wazid, A. K. Das, N. Kumar, A. G. Reddy, E. Yoon, K. Yoo, Secure signature-based authenticated key establishment scheme for future iot applications, IEEE Access 5 (2017) 3028–3043.

[15] A. Karati, S. H. Islam, M. Karuppiah, Provably secure and lightweight certificateless signature scheme for iiot environments, IEEE Transactions on Industrial Informatics 14 (8) (2018) 3701–3711.

[16] M. N. Aman, M. H. Basheer, B. Sikdar, Data provenance for iot with light weight authentication and privacy preservation, IEEE Internet of Things Journal 6 (6) (2019) 10441–10457.

[17] M. Rana, A. Shafiq, I. Altaf, M. Alazab, K. Mahmood, S. A. Chaudhry, Y. B. Zikria, A secure and lightweight authentication scheme for next generation iot infrastructure, Computer Communications 165 (1920) 85–96.

[18] P. Vijayakumar, M. S. Obaidat, M. Azees, S. H. Islam, N. Kumar, Efficient and secure anonymous authentication with location privacy for iot-based wbans, IEEE Transactions on Industrial Informatics 16 (4) (2019) 2603–2611.

[19] S. A. Chaudhry, T. Shon, F. Al-Turjman, M. H. Alsharif, Correcting design flaws: An improved and cloud assisted key agreement scheme in cyber physical systems, Computer Communications 153 (2020) 527–537. URL https://doi.org/10.1016/j.comcom.2020.02.025

[20] M. Tanveer, A. H. Zahid, M. Ahmad, A. Baz, H. Alhakami, Lake-iod: Lightweight authenticated key exchange protocol for the internet of drone environment, IEEE Access 8 (2020) 155645–155659. doi:10.1109/ACCESS.2020.3019367.

[21] B. Bera, D. Chattaraj, A. K. Das, Designing secure blockchain-based access control scheme in iot-enabled internet of drones deployment, computer communications, Volume 153 (2020) 229–249. URL https://doi.org/10.1016/j.comcom.2020.02.011

[22] M. Gharibi, R. Boutaba, S. L. Waslander, Internet of drones, IEEE Access 4 (2016) 1148–1162.

[23] T. Eisenbarth, T. Kasper, A. Moradi, C. Paar, M. Salmasizadeh, M. Shalmani, On the power of power analysis in the real world: A complete break of the keeloq code hopping scheme, in: D. Wagner (Ed.), Advances in Cryptology, CRYPTO 2008, Vol. 5157 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2008, pp. 203–220. doi:10.1007/978-3-540-85174-5_12.
URL http://dx.doi.org/10.1007/978-3-540-85174-5_12

[24] X. Cao, S. Zhong, Breaking a remote user authentication scheme for multi-server architecture, Communications Letters, IEEE 10 (8) (2006) 580–581. doi:10.1109/LCOMM.2006.1665116.

[25] D. S. Gupta, S. H. Islam, M. S. Obaidat, P. Vijayakumar, N. Kumar, Y. Park, A provably secure and lightweight identity-based two-party authenticated key agreement protocol for iiot environments, IEEE Systems Journal (2020).

[26] S. A. Chaudhry, Correcting "palk: Password-based anonymous lightweight key agreement framework for smart grid", International Journal of Electrical Power & Energy Systems 125 (2021) 106529. doi:10.1016/j.ijepes.2020.106529.

[27] S. A. Chaudhry, K. Yahya, F. Al-Turjman, M. H. Yang, A secure and reliable device access control scheme for iot based sensor cloud systems, IEEE Access 8 (2020) 139244–139254. doi:10.1109/ACCESS.2020.3012121.

[28] D. He, N. Kumar, H. Wang, L. Wang, K.-K. R. Choo, A. Vinel, A provably-secure cross-domain handshake scheme with symptoms-matching for mobile healthcare social network, IEEE Transactions on Dependable and Secure Computing 15 (4) (2016) 633–645.

[29] Z. Ali, S. A. Chaudhry, K. Mahmood, S. Garg, Z. Lv, Y. B. Zikria, A clogging resistant secure authentication scheme for fog computing services, Computer Networks 185 (2021) 107731. doi:10.1016/j.comnet.2020.107731.

[30] T. S. Messerges, E. A. Dabbish, R. H. Sloan, Examining smart-card security under the threat of power analysis attacks, Computers, IEEE Transactions on 51 (5) (2002) 541–552.

[31] P. Kocher, J. Jaffe, B. Jun, Differential power analysis, in: Advances in Cryptology CRYPTO 99, Springer, 1999, pp. 388–397.

[32] M. Abdalla, P. A. Fouque, D. Pointcheval, Password-based authenticated key exchange in the three-party setting, in: 8th International Workshop on Theory and Practice in Public Key Cryptography, PKC-05, Lecture Notes in Computer Science, vol. 3386, Les DiableretsSwitzerland, 2005, pp. 65–84.

[33] A. Irshad, M. Usman, S. A. Chaudhry, A. K. Bashir, A. Jolfaei, G. Srivastava, Fuzzy-in-the-loop-driven low-cost and secure biometric user access to server, IEEE Transactions on Reliability (2020) 1–12doi:10.1109/TR.2020.3021794.

[34] C. C. Chang, H. D. Le, A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks, ieee trans, Wireless Commun. 15 (1) (2016) 357–366.

[35] S. Malani, J. Srinivas, A. K. Das, K. Srinathan, M. Jo, Certificate-based anonymous device access control scheme for iot environment, IEEE Internet of Things Journal 6 (6) (2019) 9762–9773.

[36] H.-F. Huang, A novel access control protocol for secure sensor networks, Computer Standards & Interfaces 31 (2) (2009) 272–276.

[37] F. Li, Y. Han, C. Jin, Practical access control for sensor networks in the context of the internet of things, Computer Communications 89 (2016) 154–164.

[38] M. Luo, Y. Luo, Y. Wan, Z. Wang, Secure and efficient access control scheme for wireless sensor networks in the cross-domain context of the

iot, Security and Communication Networks 2018 (2018).

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

[40] H. H. Kilinc, T. Yanik, A survey of sip authentication and key agreement schemes, IEEE Communications Surveys & Tutorials 16 (2) (2013) 1005–1023.