liteGAP: Lightweight Group Authentication Protocol for Internet of Drones Systems

Cong Pu, *Member, IEEE*, Clare Warner, Kim-Kwang Raymond Choo, *Senior Member, IEEE*, Sunho Lim, *Senior Member, IEEE*, and Imtiaz Ahmed, *Member, IEEE*

***Abstract*—Over the past few years, the synergic usage of unmanned aerial vehicles (later drones) and Internet of Things (IoT) has successfully transformed into the Internet of Drones (IoD) paradigm, where the data of interest is gathered and delivered to the Zone Service Provider (ZSP) by drones for substantial additional analysis. Considering the sensitivity of collected information and the impact of information disclosure, information privacy and security issues should be resolved properly so that the maximum potential of IoD can be realized in the increasingly complex cyber threat environment. Ideally, an authentication and key agreement protocol can be adopted to establish secure communications between drones and the ZSP in an insecure environment. Nevertheless, a large group of drones authenticating with the ZSP simultaneously will lead to a severe authentication signaling congestion, which inevitably degrades the quality of service (QoS) of IoD systems. To properly address the above-mentioned issues, a lightweight group authentication protocol, called *liteGAP*, is proposed in this paper. *liteGAP* can achieve the authenticated key establishment between a group of drones and the ZSP concurrently in the IoD environment using lightweight operations such as hash function, bitwise XOR, and physical unclonable function (PUF). We verify *liteGAP* using AVISPA (a tool for the automatic verification of security protocols) and conduct formal and informal security analysis, proving that *liteGAP* meets all pre-defined security requirements and withstand various potential cyber attacks. Moreover, we develop an experimental framework and conduct extensive experiments on *liteGAP* and two benchmark schemes (e.g., GASE and rampIoD). Experimental findings show that *liteGAP* outperforms its counterparts in terms of computational cost as well as communication overhead.**

***Index Terms*—Information Privacy and Security, Internet of Drones, Signaling Congestion, Group Authentication.**

I. Introduction

In the third decade of the 21st century, drones have entered a new realm owing to today's technological advances in robotics automation and control, and their applications have increased rapidly in various fields. For example, on March 29, 2021, 3,281 drones lighted up night sky in Shanghai and formed into the emblem of luxury vehicle brand Genesis, making a Guinness World Records title for the most drones airborne simultaneously. To fight against coronavirus and save lives across the world, drones have been widely used by healthcare providers and biopharmaceutical companies to deliver medicines and vaccines to hard-to-reach places. Pfizer Inc. announced that Zipline (a global instant logistics company) has successfully completed the first COVID-19 vaccine drone delivery in Ghana on November 11, 2021. Drone technology also has the potential to bring huge economic and societal benefits. According to the April 2021 ''Commercial Drone Market'' report published by Grand View Research [1], the drone industry is booming and its market value is expected to be worth $47.38 billion globally by 2029. With the innovative developments in materials science, wireless communication, as well as computing and storage, it is predictable that drone technology will transform the way we work and live in the near future.

As drones are becoming more commonplace and have widespread adoption, many attempts have been made to revolutionize the traditional Internet of Things (IoT) by embracing drones, and build a promising air-ground integrated communication architecture, which is known as the Internet of Drones (IoD) [2]. The IoD paradigm partitions airspace into zones, each of which is coordinated and administered by one or more Zone Service Providers (ZSPs). The primary function of ZSPs is to allow drones to connect to a wired network. Typically, a plethora of drones are deployed to gather task-related information in the zone and deliver them to the ZSP for further information mining and analysis. With the growing prevalence of drones, numerous real-world applications have quickly emerged around the IoD, ranging from law enforcement surveillance to construction surveying and inspection. In these emerging applications, drones can often make the trip faster or accomplish a task more efficiently with less risk. A telling example is that drones can survey dangerous sites, sparing employees from exposure to threats like noxious gas or shaky structures.

A. Motivation

The transformative power of the IoD has been demonstrated in civilian applications during the coronavirus pandemic. However, to unleash the full potential of the IoD, several issues need to be properly addressed. First, the data collected by

Manuscript received xxxx xx, xxxx; revised xxxx xx, xxxx. *(Corresponding author: Cong Pu.)*

C. Pu and C. Warner are with the Department of Computer Science, Oklahoma State University, Stillwater, OK 74078, United States (e-mail: cong.pu@outlook.com, clawarn@okstate.edu).

K.K.R Choo is with the Department of Information Systems and Cyber Security, The University of Texas at San Antonio, San Antonio, TX 78249, United States (e-mail: raymond.choo@fulbrightmail.org).

S. Lim is with the Department of Computer Science, Texas Tech University, Lubbock, TX 79409, United States (e-mail: sunho.lim@ttu.edu).

I. Ahmed is with the Department of Electrical Engineering and Computer Science, Howard University, Washington, DC 20059, United States (e-mail: imtiaz.ahmed@howard.edu).

drones might contain sensitive attributes (i.e., facial recognition drones collect and store very sensitive and personally identifiable information), thus, the communications between drones and the ZSP over an open network need to be protected from unauthorized access [3]. Specifically, before exchanging sensitive information, drones and the ZSP need to perform identity verification and reach an agreement on the session keys. Second, as a result of drones' scarce resources, however, full-fledged security solutions (e.g., FPGA-based RSA and AES) cannot be directly applied [4]. We also have to be cognizant of the fact that drones' capabilities and functionalities must not be affected by the security schemes, e.g., heavyweight security schemes consume a significant amount of energy, shortening the lifespan of drones. Third, if a large group of drones communicates with the ZSP for authentication and key establishment simultaneously, a severe authentication signaling congestion will occur at the ZSP [5]. As a result, drones might be faced with authentication failure or even suffer denial of service, and the overall quality of service (QoS) is adversely affected.

B. Contribution

Based upon the above discussion, this paper proposes a security solution that allows a large group of drones to communicate with the ZSP simultaneously and securely over an open network. We also perform an in-depth security analysis and conduct extensive experiments to evaluate the proposed approach. In short, our main contributions are listed as follows:

- We propose a lightweight group authentication protocol, called *liteGAP*, for IoD systems. *liteGAP* can achieve the authenticated key establishment between a group of drones and the ZSP concurrently using lightweight operations such as hash function, bitwise XOR, and physical unclonable function (PUF).
- We verify *liteGAP* using AVISPA (a tool for the automatic verification of security protocols) [6] as well as conduct formal and informal security analysis, proving that *liteGAP* meets all pre-defined security requirements and withstand various potential cyber attacks.
- We develop an experimental framework and conduct extensive experiments on *liteGAP* and two benchmark schemes, GASE [7] and rampIoD [8], for performance evaluation.

The experimental findings demonstrate that *liteGAP* can meet all security requirements while achieving lower computational cost and communication overhead.

C. Novelty

Our work is novel in the matter of three aspects: investigating emergent IoD architecture; developing unprecedented group authentication protocol; and adopting resource-friendly operations. First, we devote our efforts to the IoD paradigm which is arguably one of the most important subjects for scientific investigation within many technical communities and commercial companies. Our thorough analysis of IoD architecture will serve as a theoretical foundation for understanding its unique security and privacy challenges and requirements. Second, we propose a group authentication protocol for IoD systems. Over the last couple of years, several authentication mechanisms have been proposed to protect the IoD communications. However, what has been lacking in the current theory is a secure and lightweight group authentication protocol that adopts resource-friendly computing operations to achieve the security and efficiency requirements concurrently for drone communications in the IoD environment. Third, the proposed group authentication protocol is realized with three resource-friendly techniques: hash function, bitwise XOR, and PUF. Compared to other heavyweight techniques (i.e., elliptic curve cryptography, bilinear pairings, etc.) which are used for resource-constrained communication systems, our solution has less computational and storage overhead while meeting the required security requirements.

D. Paper Organization

The rest of the paper is organized as follows. Section II presents and analyzes the state of the art. The background information of associated technique is provided in Section III. We describe network and adversary models, as well as security requirements in Section IV. *liteGAP* is proposed in Section V. Section VI and VII are devoted to security verification and analysis, and experimental study, respectively. Lastly, the paper is concluded in Section VIII.

II. Related Work

During recent years, some researchers have investigated authenticated key agreement mechanisms so that the entities of IoD systems are able to exchange information securely in an untrustworthy environment. In [9], the authors first expose real weaknesses (i.e., single point of failure and lack of inter-domain authentication) of centralized IoD authentication systems. Then, they design a blockchain assisted cross-domain authentication scheme to protect drone communications in the IoD environment, where a drone's federated identity is created using a threshold signature scheme. Moreover, drones from different domains are able to verify each other's identities and set up secure session keys with the assistance of smart contract. The experimental study demonstrates that the proposed scheme has promising performance from the efficiency and effectiveness point of view. However, the authentication signaling congestion problem existing in the intra-domain did not get authors' attention at all. In [8], an authentication scheme based upon elliptic curve cryptosystem and hash algorithm is developed for IoD networks. Before the drone and the user share any critical information over an insecure channel, their identifications are required to be verified first, and then they can reach an agreement on the session key. One striking feature of their approach is the privacy guarantee: the user's and drone's identity information are not disclosed during the authentication process. The proposed scheme delivers better performance along with advanced security features, nevertheless, it only allows the user to authenticate with one drone at a time. When the user needs to establish mutual authentication with a large number of drones simultaneously for the task of interest, a non-negligible amount of communication overhead

can be expected because the same authentication process has to be repeated a sufficient number of times. The authors in [10] design a group handover mechanism for 5G-enabled vehicle-to-Everything networks, where the vehicle leader performs mutual authentication with the core network on behalf of other vehicles. However, vehicles are assumed to have unlimited resources to perform cryptographic operations, which is not the case in the IoD environment.

Another area of research has been on the development of security solutions for the IoT networks. In [11], the authors investigate the device-to-device (D2D) communication and its security threats in the 5G-enabled IoT setting. To protect IoT devices (e.g., drones) from malicious attacks, a 5G D2D ProSe standard compatible authentication mechanism is proposed. Precisely, the leader drone first registers with the core network, and then broadcasts a proxy signature so that it can achieve mutual authentication with other adjacent drones. After that, the leader drone serves as a relay point between the backbone network and the drone swarm for the exchange of critical data. Regrettably, acting as the relay node will turn the leader drone into a single point of failure (SPoF), which makes the entire network vulnerable to cyber attacks. In [12], a federated learning (FL) based drone authentication model is designed for drone-enabled IoT networks, where the deep neural network integrated with stochastic gradient descent optimization is performed on drones locally for authentication. In addition, to secure critical parameters, the secure aggregation mechanism and homomorphic encryption are adopted. Unfortunately, the major drawback is that the deep learning model is energy intensive to the resource-constrained drones.

In [13], the authors propose a group signature mechanism for blockchain-enabled mobile-edge computing systems. If the new block contains a valid group signature created through the BLS aggregate signature algorithm, it is regarded as a legal block. In addition, they propose an authentication scheme for mobile device users to relocate between different groups in the network. The basic idea is to store the authentication credentials in the blackchain so that mobile device users can access them in the blockchain for authentication. In [14], a certificate-free authenticated key agreement mechanism is developed for 5G D2D networks, where the public key and elliptic curve cryptosystems are adopted to realize the authentication. Moreover, a digital signature is created to protect D2D group communications from internal attackers. The authors in [15] develop a secure message exchange protocol for IoT networks. Through the secure protocol, IoT devices and untrustworthy edge servers are able to exchange information freely. In [16], the authors point out that sequentially authenticating RFID tags will generate heavy communication workloads. To resolve this issue, a security solution that can achieve group authentication of RFID tags is developed. If a group of RFID tags respond to the authentication requests simultaneously, a confirmable bit-collision pattern will be generated, indicating that the responses of authentication requests come from the entire group. However, in all the abovementioned studies, the authors did not take into account mobility, thus, their approaches are unable to be employed for IoD systems.

In [17], the authors propose an in-network caching for fast content delivery in Vehicle-to-Grid (V2G) networks, where each vehicle will evaluate the reputation score of content provider before retrieving their content. In addition, the blockchain technique is adopted to securely store the reputation value and incentives-related transactions. In [18], the authors present blockchain and UAV-enabled edge computing based energy trading services for the V2G environment. In their approach, the electric vehicle will select a charging station which is close to its moving path with the assistance of edge node. The investigated topics in [17] [18] are urgent, but are not duplicative of what is being investigated in this paper. Given the expected impact of the research outcomes, the proposed research is expected to amplify the authors' productivity in the V2G domain, as well as be complementary to what is being done elsewhere and, more likely, to be synergistic.

In summary, many researchers spent effort on the security issues of IoD systems and developed various authentication mechanisms. However, they did not give much attention to a lightweight group authentication protocol based upon lightweight operations to protect communications between a group of drones and the ZSP in the IoD environment.

III. Preliminary: Physical Unclonable Function

Physical unclonable functions (PUFs) are universally utilized as a hardware-specific security primitive to offer cryptographic services for electronic devices [19]. The physical structure of PUF is formed in the process of manufacturing. Since it is inevitable for each integrated circuit to have slight physical differences from the manufacturing process, the PUF is believed to be impossible to replicate or clone. Thanks to its unique features, the PUF is generally considered to be the identification of an electronic device, which is analogous to a person's social security number.

Typically, the PUF is fed with an input, termed *challenge*, and generates an output, named *response*. The combination of challenge and response goes by the name challenge-response pair (CRP). A single PUF always responds to the same challenge equivalently (i.e., the same response is produced), and two distinct PUF instances should respond to the same unbiased challenges differently (i.e., the different responses are produced). According to [20], the PUF can be represented as a mathematical function, denoted as $res = F_{puf}(che)$, where *che* and *res* indicate PUF's challenge and response, respectively.

In noisy environments, the identical challenges fed to the PUF might not be able to get the same responses [20]. In other words, the PUF is sensitive to external environment changes/noise, thus, the secret data of cryptographic operations might not be regenerated by the PUF. To resolve this important issue, error correction code (ECC) and fuzzy extractor can be integrated with the PUF. First, we define an algorithm to generate the response, *rGen*. The *rGen* algorithm will output a set $\{res, S\}$. Here, *res* is the CRP response, which is the value to be regenerated by the PUF. *S* is a helper string which is fed into the PUF to regenerate the CRP response *res*. The error correction code (ECC) [21] is adopted to eliminate up to *x* bit errors in the CRP response *res*.

Algorithm 1: Response Generation Algorithm *rGen*

Input: Modulus n; Challenge *che*

Function rGen (*n, che*) **:**

```
/* ←⊛ denotes sampling                          */
/* ⊕ denotes exclusive OR function              */
/* Z_n denotes the set of remainders in
   arithmetic modulo n                          */
```

$O = F_{puf}(che)$;

$res \xleftarrow{\circledast} \mathbb{Z}_n$;

$S = O \oplus ECC(res)$;

return $\{res, S\}$;

Algorithm 2: Response Restore Algorithm *rRes*

Input: Challenge *che*; Helper string S

Function rRes (*che, S*) **:**

$O^{'} = F_{puf}(che)$;

$res = D_{er}(S \oplus O^{'})$;

return *res*;

We also design a response restore algorithm, denoted as $rRes$. The main purpose of $rRes$ is to allow the PUF to restore the CRP response *res* with the assistance of the helper string S and the error decoding algorithm D_{er}, even if the PUF produces an output $O^{'}$ that differs from the original output O by at most x bits.

IV. Network and Adversary Models

A. Network Model

In this paper, there are two major entities, ZSP and drones, which are shown in Fig. 1. It is assumed that every drone is furnished with a PUF, and has limited resources (e.g., battery energy). The focus of this paper is on the mutual authentication with privacy protection in the IoD environment, thus, we do not spend effort on the design and creation of real PUF. For simplicity, the PUF is simulated as a secure process integrated with fuzzy extractor method and error-correcting technique (see more details about the implementation of PUF in Section III). In addition, the ZSP is regarded as a trusted entity with no resource constraints.

Without loss of generality, we consider the scenario in which a large group of drones wants to exchange sensitive information with the ZSP. Since the data will be transmitted over an insecure communication channel, thus, drones need to authenticate and establish secure session keys with the ZSP before sharing any critical information. However, if a large group of drones sends their separate authentication request messages to the ZSP simultaneously, authentication signaling congestion might occur at the ZSP, which can cause authentication failure or even denial of service. Thus, an efficient and lightweight group authentication protocol dedicated for resource-constrained drones is required for IoD systems.

B. Adversary Model

The formalization of the adversary model is based on Dolev–Yao threat framework [22]. Thus, the adversary is believed to have boundless power so that it can control the communication network. In addition, the adversary can

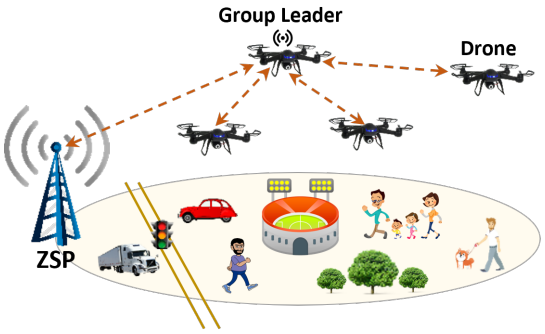


Fig. 1. Network model.

disguise itself as a legitimate entity of the network. This, in turn, means that the adversary has the ability to compromise the transmitted messages. In the IoD setting, it is difficult or impossible to protect drones physically. Thus, there is some chance that the drone is physically captured by the adversary [23]. However, if the adversary plans to fetch the secret data from the PUF, the physical characteristics of PUF will be compromised and the original CRP will be destroyed. To sum up, the primary goal of the adversary is to pretend to be a legitimate entity and communicate with the trusted ZSP or any legitimate drone, and then cause serious damage to the national interest, organizations or individuals.

C. Security Requirements

According to the well-known security objectives of computing services [24], we outline the following security requirements to be satisfied by our approach *liteGAP*.

- Group Authentication: *liteGAP* must ensure the authenticity of a group of drones and the ZSP, that is, each is the entity that it claims to be. Additionally, *liteGAP* should not allow any adversary to disguise itself as a legitimate entity (drone or ZSP) for malicious purposes.
- Group Session Key Establishment: After successful group authentication, *liteGAP* must achieve an agreement on session keys between a group of drones and the ZSP. In addition, *liteGAP* must ensure that an adversary is unable to obtain intelligence from the captured session key.
- Confidentiality: After a secure session key is established, *liteGAP* must assure that confidential messages are securely shared between drones and the ZSP over an open network, and not made available or disclosed to unauthorized adversary.
- Integrity: *liteGAP* must perform the verification on the source of messages, and make sure that the messages are free from deliberate or inadvertent unauthorized manipulation or modification.
- Anonymity: *liteGAP* must use the pseudonyms of drones during the group authentication phase. Moreover, *liteGAP* must ensure that the real identities of drones are only known to the trusted ZSP, and an adversary cannot reveal drones' real identities via eavesdropping.
- Secure Against Cyber Attacks: *liteGAP* must be secure against well-known cyber attacks such as ZSP spoofing attack, replay attack, message modification attack, man-

TABLE I
NOTATIONS

Notation	Meaning
ZID_s	The identity of ZSP Z_s
ID_i	The real identity of drone N_i
PID_i	The pseudonym of drone N_i
che_i	Drone N_i's PUF challenge
res_i	Drone N_i's PUF response
(che_i, res_i)	Drone N_i's challenge-response pair (CRP)
$F^i_{puf}(\cdot)$	Drone N_i's PUF
$rGen(\cdot)$	PUF response generation algorithm
$rRes(\cdot)$	PUF response restore algorithm
S	Helper string
n	Modulus n
$H(\cdot)$	Hashing algorithm
$\oplus$	Bitwise XOR
$\|$	Concatenation operation
T_j	The *j*th task or mission
t	Timestamp
r^t_i and r^t_s	Nonce generated by drone N_i and ZSP Z_s
PR_s	ZSP Z_s's private key
GID_j	Group identity for drones associated with T_j
GT_j	Group token for drones associated with T_j
GK_j	Group key for drones associated with T_j
GL_j	Group leader for T_j
$C(\cdot)$	Message authentication code (MAC) function
M^k_{id}	The message k generated by entity *id*
MAC^k_{id}	The MAC of message k generated by entity *id*
MAC^*_j	The aggregate MAC for T_j
M^*_j	The aggregate authentication request for T_j
$SK_{i,s}$	Secret session key between drone N_i and ZSP Z_s

in-the-middle attack, drone capture attack, known session key attack, and drone impersonation attack.

V. *liteGAP*: LIGHTWEIGHT GROUP AUTHENTICATION SCHEME

We propose a lightweight group authentication protocol, also called *liteGAP*, for IoD systems. In general, *liteGAP* is designed based upon lightweight operations such as bitwise XOR, hashing, and PUF operations. The basic idea of liteGAP is that the system is first initialized through ZSPs choosing and publishing a set of system parameters and functions, and drones selecting their real identities and PUF challenges. After system initialization, each drone can register with the ZSP by exchanging group authentication information and identity information. Finally, the ZSP will authenticate a group of drones and establish secure session keys with all drones simultaneously. Specifically, *liteGAP* is comprised of three stages to achieve the authenticated key establishment between a group of drones and the ZSP concurrently: (i) setup stage; (ii) registration stage; as well as (iii) group authentication and key establishment stage. In this paper, we choose drone N_i as a representative example to explain the operations in *liteGAP*. We also assume that a total of p drones, including drone N_i ($i \leq p$), are deployed for the task/mission T_j. The mathematical symbols used in *liteGAP* and their meaning are provided in Table I.

A. Setup Phase

In the setup phase, ZSP Z_s chooses system parameters and secure function, and drone N_i chooses its identity-related information. When the setup phase is over, Z_s will have a private key, while N_i will obtain its real identity and PUF challenge. The detailed steps are as follows.

1) Z_s chooses its identity ZID_s, private key PR_s, and a hashing algorithm $H(\cdot)$. Z_s publishes $(ZID_s, H(\cdot))$ while keeps PR_s securely.
2) N_i randomly chooses its real identity ID_i and PUF challenge che_i. N_i saves (ID_i, che_i) secretly.

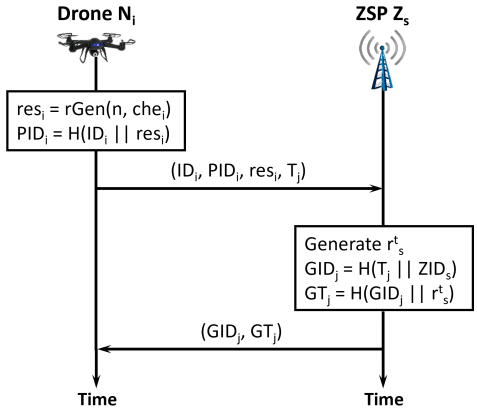


Fig. 2. Registration phase.

B. Registration Phase

In the registration phase, drone N_i registers with ZSP Z_s by submitting a registration request. Upon receiving the registration request from N_i, Z_s generates the group authentication related information and share them with N_i. When the registration phase is over, Z_s will obtain N_i's real identity, pseudonym, PUF challenge, and group authentication information, while N_i will receive its group authentication related information. Fig. 2 presents the registration process of *liteGAP*, and the key steps of registration process are explained below.

1) N_i feeds its PUF challenge che_i into response generation algorithm $rGen(\cdot)$ and computes the corresponding PUF response $res_i = rGen(n, che_i)$. N_i also feeds its ID_i and res_i into the hashing algorithm to calculate the pseudonym $PID_i = H(ID_i \| res_i)$. Here, the pseudonym of drone, rather than the real identity of drone, will be used for the communication with the group leader later to preserve the identity privacy of drone.
2) N_i sends a registration request containing $(ID_i, PID_i, res_i, T_j)$ to Z_s via a secure channel (e.g., time-based OTP algorithm (TOTP) [25]).
3) After receiving the registration request from N_i, Z_s generates a nonce r^t_s, and computes the group identity $GID_j = H(T_j \| ZID_s)$ and the group token $GT_j = H(GID_j \| r^t_s)$. Note that this step is only executed when Z_s receives the first registration request for the task T_j.
4) Z_s stores $(ID_i, PID_i, res_i, T_j, GID_j, GT_j)$ in the database, and sends (GID_j, GT_j) to N_i via a secure channel. Here, Z_s stores GID_j and GT_j so that it does not need to re-calculate them when the other drone registers for the same task T_j.

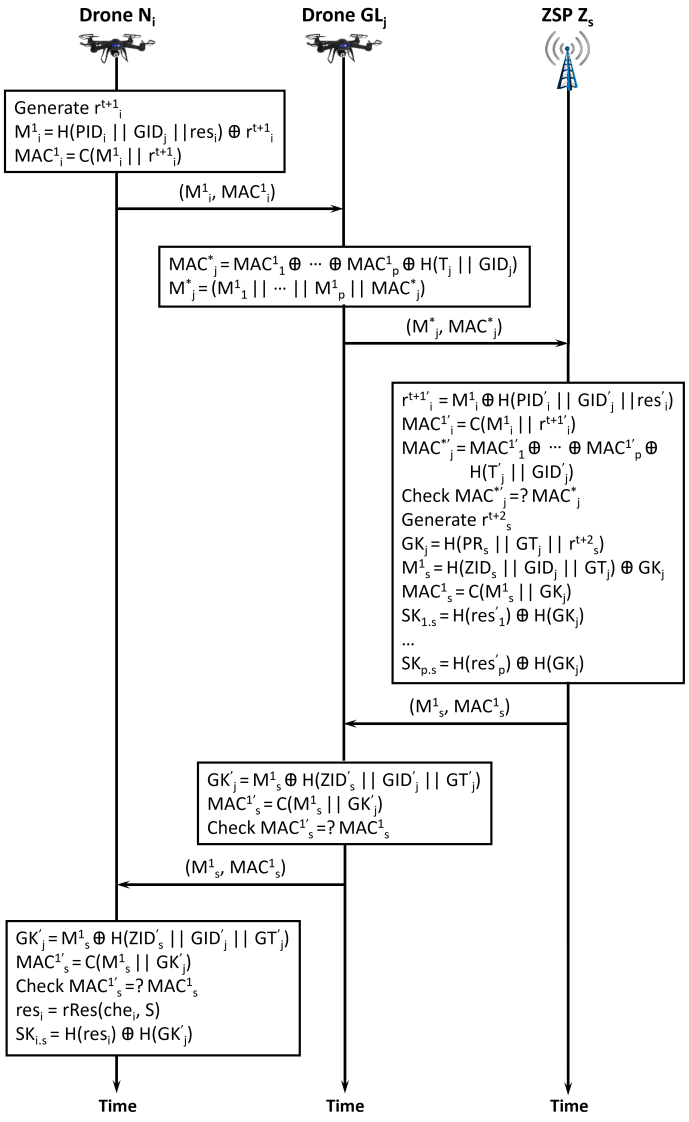


Fig. 3. Group authentication and key establishment phase.

5) N_i receives (GID_j, GT_j) and stores them along with (ID_i, che_i) securely.

C. Group Authentication and Key Establishment Phase

During the process of group authentication and key establishment, ZSP Z_s authenticates a pack of drones and reaches an agreement on the secret session keys with all drones in the group simultaneously. Fig. 3 presents the group authentication and key establishment process of *liteGAP*, and its major steps are explained below.

1) A group leader GL_j is selected based on fuzzy logic algorithm which takes input as drones' residual energy and the distances between drones and ZSP Z_s [26].
2) N_i generates a nonce r_i^{t+1} and calculates $M_i^1 = H(PID_i \parallel GID_j \parallel res_i) \oplus r_i^{t+1}$. N_i also calculates a message authentication code (MAC) $MAC_i^1 = C(M_i^1 \parallel r_i^{t+1})$. Finally, N_i sends its authentication request (M_i^1, MAC_i^1) to GL_j.
3) When GL_j receives all authentication requests from the group, it calculates an aggregate message authentication code [27] $MAC_j^* = MAC_1^1 \oplus MAC_2^1 \oplus \cdots \oplus MAC_p^1 \oplus H(T_j \parallel GID_j)$. Here, p is the total number of drones in the group.
4) GL_j generates an aggregate authentication request $M_j^* = (M_1^1 \parallel M_2^1 \parallel \cdots \parallel M_p^1 \parallel MAC_j^*)$, and sends ($M_j^*$, MAC_j^*) to Z_s.
5) After Z_s receives (M_j^*, MAC_j^*) from GL_j, it retrieves drones' identity-related information and group authentication related information from the database. Then, it restores the nonce and computes the MAC for the authentication request from each drone as the following.
$$r_i^{t+1'} = M_i^1 \oplus H(PID_i^{'} \parallel GID_j^{'} \parallel res_i^{'})$$
$$MAC_i^{1'} = C(M_i^1 \parallel r_i^{t+1'})$$
Here, Z_s can easily relate M_i^1 to the corresponding PUF response $res_i^{'}$. This is because Z_s obtains N_i real identity, pseudonym, PUF challenge, and group authentication information and stores them in the database during the registration phase.
6) Z_s computes $MAC_j^{*'} = MAC_1^{1'} \oplus MAC_2^{1'} \oplus \cdots \oplus MAC_p^{1'} \oplus H(T_j^{'} \parallel GID_j^{'})$, and checks $MAC_j^{*'} \stackrel{?}{=} MAC_j^*$. If they are equal, Z_s proceeds with the following steps. Otherwise, the group authentication request is rejected.
7) Once the verification succeeds, Z_s generates a nonce r_s^{t+2} and calculates the group key $GK_j = H(PR_s \parallel GT_j \parallel r_s^{t+2})$. It also generates M_s^1 and MAC_s^1 and sends (M_s^1, MAC_s^1) to GL_j.
$$M_s^1 = H(ZID_s \parallel GID_j \parallel GT_j) \oplus GK_j.$$
$$MAC_s^1 = C(M_s^1 \parallel GK_j)$$
Then, Z_s establishes secure session keys for all drones in the group as the following.
$$SK_{1.s} = H(res_1^{'}) \oplus H(GK_j)$$
$$SK_{2.s} = H(res_2^{'}) \oplus H(GK_j)$$
$$\cdots$$
$$SK_{p.s} = H(res_p^{'}) \oplus H(GK_j)$$
Here, since Z_s will generate a different nonce each time, the calculated group key will also be different. As a result, Z_s is able to establish different secure session keys with all drones during the group authentication and key establishment stage. In other words, the secure session keys will be frequently updated in our approach *liteGAP*.
8) On receiving (M_s^1, MAC_s^1) from Z_s, GL_j retrieves the group authentication related information and calculates the following.
$$GK_j^{'} = M_s^1 \oplus H(ZID_s^{'} \parallel GID_j^{'} \parallel GT_j^{'})$$
$$MAC_s^{1'} = C(M_s^1 \parallel GK_j^{'})$$
After that, GL_j checks the validation of $MAC_s^{1'} = MAC_s^1$. If they are equal, GL_j broadcasts (M_s^1, MAC_s^1) to all drones in the group. Otherwise, it discards the message.
9) After receiving (M_s^1, MAC_s^1) from GL_j, N_i first restores $GK_j^{'} = M_s^1 \oplus H(ZID_s^{'} \parallel GID_j^{'} \parallel GT_j^{'})$, and then verifies whether $MAC_s^{1'}$ equals to MAC_s^1 or not (similar

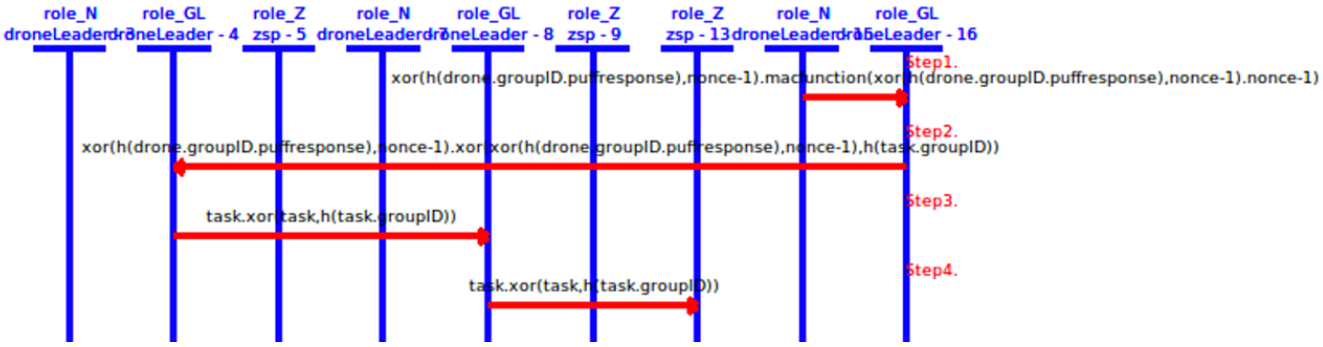
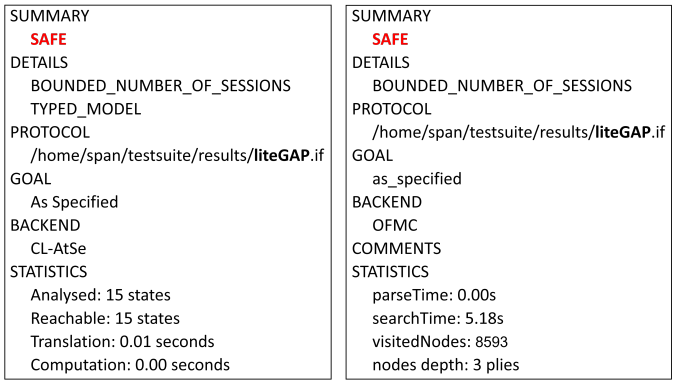


Fig. 4. Communication sequence diagram of AVISPA security verification.



(a) (b)

Fig. 5. Security verification results using AVISPA's CL-AtSe and OFMC.

to the previous step). If the verification fails, N_i discards the message. If the verification succeeds, it computes its own secure session key as follows.

$res_i = rRes(che_i, S)$

$SK_{i.s} = H(res_i) \oplus H(GK_j^{'})$

By this time, the group authentication between all drones in the group and ZSP Z_s is executed completely, and secure session keys have been successfully established for all drones to securely communicate with ZSP Z_s.

VI. *liteGAP*'s Resilience and Security Analysis

A. Security Verification

Security protocols might have weaknesses which can be exploited by the adversary to conduct serious attacks without compromising cryptography, such as masquerading attacks or replay attacks. Thus, we choose AVISPA [28], a tool for the automatic verification of security protocols, to automatically analyze and validate our approach *liteGAP*, and demonstrate that *liteGAP* is able to work securely even under worst-case adversarial environments. Typically, the to-be-validated security protocol can be represented as a security problem in the HLPSL (the programming language on AVISPA) [6], and then evaluated against masquerading attacks, replay attacks, and other unknown attacks on AVISPA. If the security protocol suffers from a specific attack, AVISPA will display the vulnerable scenario as a sequence diagram. Otherwise, the security protocol is marked as ''safe'' by AVISPA.

AVISPA provides two evaluation components: On-the-fly Model-Checker (OFMC) and Constraint-Logic-based Attack Searcher (CL-AtSe). Specially, OFMC evaluates the security protocol through falsification and bounded verification. CL-AtSe is able to deal with algebraic properties of cryptographic operators and associativity of message concatenation, as well as detect type-flaw attacks. In Virtual Box [29], we first install Ubuntu 10.04, and then set up and configure a fully-functional SPAN+AVISPA [6] environment. The communication sequence diagram of AVISPA security verification as well as the outputs of OFMC and CL-AtSe are demonstrated in Fig. 4 and 5 respectively, showing that *liteGAP* is a safe security protocol and secure against masquerading attacks, replay attacks, and other unknown attacks. The HLPSL security verification programs are available at the https://github.com/congpu/liteGAP.

B. Formal Security Analysis

We provide a formal security analysis of *liteGAP*, which focuses on sharing the secret information res_i between drone ID_i and ZSP Z_s. Here, the secret information res_i is used to achieve the authentication and key establishment between drone ID_i and ZSP Z_s. The goal of formal security analysis is to prove that res_i is a good shared secret between drone ID_i and ZSP Z_s. In other words, the secret information res_i should not be accessed by any attacker. We adopt the inference rules proposed by Mao and Boyd [30] to build the formal security analysis of the secret information res_i in *liteGAP*. Moreover, according to the operations in setup and registration phases, we can build the following beliefs.

1) $ID_i|\equiv ID_i \overset{res_i}{\longleftrightarrow} Z_s$ and $Z_s|\equiv Z_s \overset{res_i}{\longleftrightarrow} ID_i$: The initial response res_i of drone ID_i is securely shared between drone ID_i and ZSP Z_s.
2) $ID_i|\equiv Z_s \triangleleft \| ID_i$: The real identify of drone ID_i is known by ZSP Z_s.
3) $ID_i|\equiv ID_i \overset{PID_i}{\longleftrightarrow} Z_s$ and $Z_s|\equiv Z_s \overset{PID_i}{\longleftrightarrow} ID_i$: ZSP Z_s saves the pseudonym of drone ID_i in its database, while drone ID_i can compute its PID_i using its real identify and response R_i^t.
4) $ID_i|\equiv ID_i \overset{GID_j}{\longleftrightarrow} Z_s$ and $Z_s|\equiv Z_s \overset{GID_j}{\longleftrightarrow} ID_i$: The group identity GID_j is securely shared between drone ID_i and ZSP Z_s.
5) $ID_i|\equiv ID_i \overset{GT_j}{\longleftrightarrow} Z_s$ and $Z_s|\equiv Z_s \overset{GT_j}{\longleftrightarrow} ID_i$: The group token GT_j is securely shared between drone ID_i and ZSP Z_s.
6) $ID_i|\equiv Z_s \triangleleft \| r^{t+1}$ and $Z_s|\equiv ID_i|\equiv \{Z_s\} \triangleleft \| r^{t+1}$: Drone ID_i generates a new r^{t+1} each time.

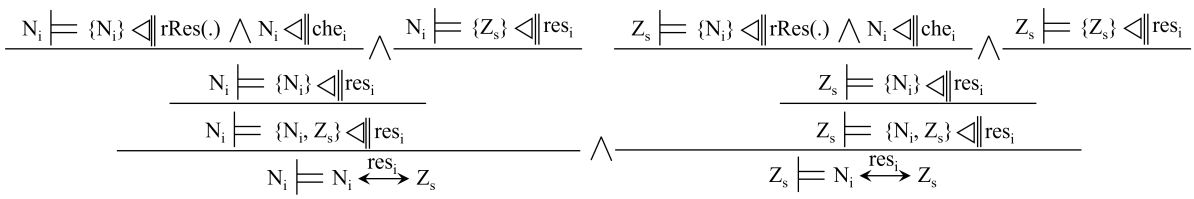


Fig. 6. Proof that drone ID_i and ZSP Z_s believe that secret information res_i is only shared between themselves.

7) $ID_i \models \#(res_i)$: Drone ID_i generates a fresh res_i each time.

Fig 6 demonstrates the formal security analysis of *liteGAP* with regards to sharing the secret information res_i between drone ID_i and ZSP Z_s. The rationale behind proving the security of secret information res_i is that res_i is critical for the authentication as well as the establishment of session key. We first establish the statements, $ID_i \models ID_i \overset{res_i}{\longleftrightarrow} Z_s$ and $Z_s \models ID_i \overset{res_i}{\longleftrightarrow} Z_s$, and make them become the foundation of the logical proof. Second, the Good Key rule [30] is applied to the statements $ID_i \models ID_i \overset{res_i}{\longleftrightarrow} Z_s$ and $Z_s \models ID_i \overset{res_i}{\longleftrightarrow} Z_s$, respectively. The Good Key rule indicates that if ID_i believes that res_i is only available to ID_i and Z_s ($ID_i \models \{ID_i, Z_s\} \triangleleft \| res_i$ or $Z_s \models \{ID_i, Z_s\} \triangleleft \| res_i$), and ID_i knows that res_i is fresh ($ID_i \models \#(res_i)$), then ID_i believes that res_i is a good shared secret information between ID_i and Z_s. Third, we apply the Confidentiality rule [30] to prove $ID_i \models \{ID_i, Z_s\} \triangleleft \| res_i$ and $Z_s \models \{ID_i, Z_s\} \triangleleft \| res_i$. Before that, we need to show that res_i is a shared secret information between ID_i and Z_s ($ID_i \models ID_i \overset{res_i}{\longleftrightarrow} Z_s$ and $Z_s \models Z_s \overset{res_i}{\longleftrightarrow} ID_i$). Fortunately, the above statement is one of the initial beliefs, thus, the truth of the security claims $ID_i \models ID_i \overset{res_i}{\longleftrightarrow} Z_s$ and $Z_s \models ID_i \overset{res_i}{\longleftrightarrow} Z_s$ are successfully proved. In summary, according to Fig 6, it is strongly believed that the secret information res_i is a good shared secret between drone ID_i and ZSP Z_s.

C. Informal Security Analysis

In the following, we will exhibit how *liteGAP* satisfies the pre-defined security requirements in Section IV.C. First of all, *liteGAP* can achieve group authentication between a group of drones and the ZSP. This is because the group leader drone first collects separate authentication request from a group of drones. Then, it generates and issues an aggregate authentication request to the ZSP. After that, the ZSP can verify the identity of each drone based on the separate authentication request. Moreover, a group of drones and the ZSP can reach an agreement on the secret session keys for future communications using *liteGAP*. After the group authentication succeeds, the ZSP will calculate the group key which will be utilized to produce the secret session key for each drone. Third, *liteGAP* can guarantee the confidentiality of messages exchanged in an open network because the messages are encrypted using the established session key. Fourth, *liteGAP* can achieve integrity because a message authentication code (MAC) is generated for each exchanged message. Last but not least, *liteGAP* supports anonymous communication for IoD systems. The rationale is that each drone is using its pseudonym, rather than its real identity, in the exchanged messages.

liteGAP is also secure against many well-known security attacks. First, *liteGAP* can defend against physical capture attack. The attacker might capture drone N_i and retrieve its identity-related information stored in the memory, e.g., ID_i and che_i, through probing attack. However, the attacker is unable to obtain the critical cryptography-related information such as res_i from drone N_i. This is because the PUF of N_i will be destroyed and the original res_i cannot be regenerated whenever the attacker plans to retrieve res_i from N_i's integrated circuit. Second, *liteGAP* is secure against replay attack. Since each message is piggybacked with timestamp ts, ZSP Z_s is able to verify ts, and then detect and discard the replayed messages. Third, *liteGAP* is secure from drone impersonation attack. Suppose that an adversary wants to impersonate a legitimate drone N_i in order to establish authentication with ZSP Z_s to cause some financial and strategic damages. In order to send a valid authentication request to the group leader GL_j, e.g., (M_i^1, MAC_i^1) on behalf of legitimate drone N_i, the adversary obtains the identifier of group leader GL_j and then generates a nonce r_i^{t+1}. However, without having the valid CRP (che_i, res_i) of legitimate drone N_i, it is a difficult task for the adversary to generate the valid M_i^1 and MAC_i^1 which can be correctly decoded by ZSP Z_s. As a result, the adversary cannot generate a valid authentication request on behalf of legitimate drone N_i. Thus, it is impractical for the adversary to impersonate drone N_i. Fourth, *liteGAP* generates message authentication code MAC to authenticate the corresponding message, thus, *liteGAP* can defend against message modification attack. Lastly, *liteGAP* is not vulnerable to man-in-the-middle attack. In *liteGAP*, ZSP Z_s first authenticates and establishes a secure session key with drone N_i. After that, Z_s and N_i can safely communicate over an insecure channel, and the attacker is unable to replay and modify the transmitted messages. In summary, *liteGAP* can meet all required security requirements as outlined in Section IV.C., and the list of achieved security requirements is summarized in Table. II.

VII. Performance Evaluation

A. Experimental Environment

We build an experimental framework on the MacBook Air laptop and conduct extensive experiments to evaluate the performance of *liteGAP*. An Eclipse simulation environment [31] is set up on the MacBook Air laptop, where *liteGAP* and two benchmark schemes are implemented in Java programming language. The MacBook Air laptop runs macOS Ventura 13.3.1 operating system with Apple M2 chip (8-core CPU, 10-core GPU, and 16-core Neural Engine), and the size of unified memory and SSD hard drive are 8GB and 512GB, respectively.

TABLE II
ACHIEVED SECURITY REQUIREMENTS

Security Requirement	*liteGAP*
Group Authentication	Yes
Group Session Key Establishment	Yes
Confidentiality	Yes
Integrity	Yes
Anonymity	Yes
Replay Attack	Yes
Man-In-The-Middle Attack	Yes
Drone Capture Attack	Yes
Drone Impersonation Attack	Yes
Message Modification Attack	Yes
Known Session Key Attack	Yes
ZSP Spoofing Attack	Yes

B. Benchmark Schemes and Performance Metrics

We choose two representative protocols, GASE [7] and rampIoD [8], as the benchmark schemes, and compare them with *liteGAP* for performance evaluation and analysis. rampIoD represents the typical authentication schemes which have been widely proposed in the IoD community, where a central authority authenticates the two entities (i.e., user and drone), and then helps them mutually authenticate each other and establish a session key. Since the current IoD community does not have similar group authentication technique, we have to select a group authentication protocol from a similar environment as another benchmark scheme. GASE is a group authentication protocol with key agreement feature which is proposed for edge computing environments. The basic idea of GASE and rampIoD are presented below:

- GASE: The objective of GASE is to validate and authenticate a mass of IoT devices without overburdening the central server in the cloud-edge-IoT environment. First, all registered IoT devices are divided into a number of groups, including one edge node and one group leader. Second, all members in the group are authenticated by the group leader using multi-secret sharing scheme. Third, the group leader sends the validated IoT devices' identifiers to the edge node. Lastly, the edge node combines all identifiers and transmits it to the central server for verification.
- rampIoD: rampIoD is designed to establish an authenticated communication between the user and the drone in the IoD environment. First, the IoD system is initialized through choosing and publishing the system parameters by the control room. Second, the drone and the user register themself with the control room to obtain their secret credentials. Third, before the user and the drone can exchange any sensitive information securely, they are required to achieve the authentication with the control room first. Finally, with the assistance of the control room, the user and the drone will establish a secure session key for future communication.

The performance of *liteGAP*, GASE, and rampIoD are measured in diverse performance metrics such as communication cost, run time, CPU time, as well as storage overhead. In the following, we provide the meaning of performance metric and explain how to measure and obtain the corresponding results.

TABLE III
COMPARISON OF COMMUNICATION COST

Protocol	No. of Trans. Msg	Energy Consumption of Trans.
liteGAP	203	2.29×10^{-2}
GASE	403	4.54×10^{-2}
rampIoD	600	6.76×10^{-2}

*In this experiment, we assume that 200 drones form a group and want to authenticate and establish secure session keys with the ZSP simultaneously.

- The communication cost is represented in terms of two sub-metrics, which are the number of transmitted messages and the energy consumption of message transmissions. Since the actual wireless communication between the IoD entities is not being simulated in the experiments, we just simply investigate *liteGAP* and two benchmark schemes, and count the number of transmitted messages. The energy consumption of message transmissions is the product of the number of transmitted messages and the energy consumption of sending and receiving a single message [32].
- The run time and the CPU time are very similar; both of them are measuring the total time elapsed from when the protocol begins execution to when the protocol finishes execution. However, the major difference between the run time and the CPU time is that the CPU time does not include the latency due to operating in low-power idle state as well as input/output operation delay.
- The storage overhead indicates how much memory space is required by the protocol.

We choose two experimental parameters, which are the number of drones in the network and the number of algorithm executions, to measure the results of performance metrics. The reason that we select the number of drones in the network as one of the experimental parameters is because this paper focuses on the group authentication protocol. By varying the number of drones in the network, we can easily observe the performance difference between our approach *liteGAP* and benchmark scheme GASE and rampIoD, and how much performance improvement our approach *liteGAP* can make. We also obtain the experimental results by changing the number of algorithm executions, which will help us observe the performance of all three schemes from a long-term running point of view.

C. Performance Results and Analysis

First, the communication cost is presented in Table. III, where the number of transmitted messages and the energy consumption of message transmissions are obtained for *liteGAP*, GASE, and rampIoD. In this experiment, we assume that 200 drones form a group and want to authenticate and establish secure session keys with the ZSP simultaneously. According to the communication sequence diagram of rampIoD, a total of 600 messages are required to be transmitted when a group of 200 drones are considered. In fact, each drone is required to exchange three (3) messages with the ZSP to successfully complete the process of authentication and key agreement in rampIoD. In GASE, a total of 403 messages are needed when a group of 200 drones exist in the network. First,

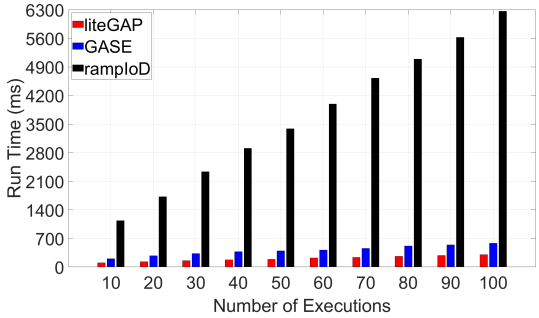


Fig. 7. The run time against the number of executions.

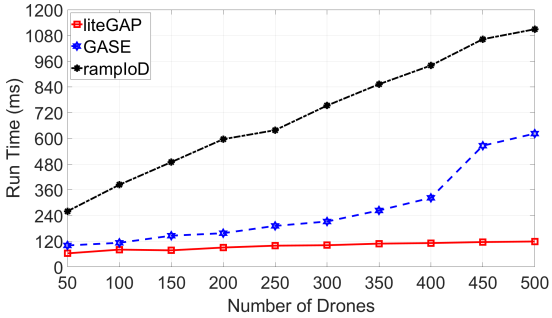


Fig. 8. The run time against the number of drones.

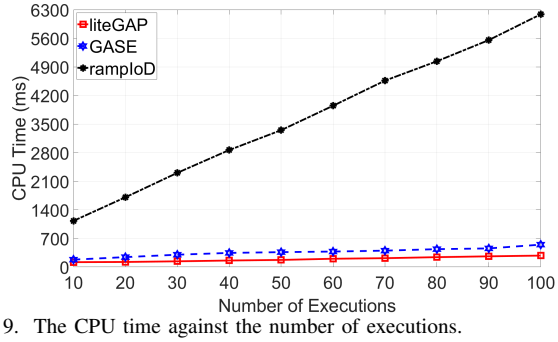


Fig. 9. The CPU time against the number of executions.

each drone sends a message to the group leader drone for revealing its secret share, and then the group leader drone broadcasts one confirmation message to all drones. After that, each drone sends a message with the MAC to the group leader drone. Finally, the group leader drone transmits two messages to the authentication server for authenticating the group. In summary, a total of 403 messages are transmitted during the entire process. However, our approach *liteGAP* only needs 203 messages exchanged between a group of 200 drones and the ZSP. To be specific, the group leader drone first receives an independent authentication request message from each drone in the group. After that, two additional messages (e.g., aggregate authentication request and authentication response messages) will be exchanged between the group leader drone and the ZSP. Finally, the authentication response message is broadcasted to all other drones in the group by the group leader drone. To sum up, 203 message transmissions are observed by *liteGAP*. In addition, the energy consumption of message transmissions for *liteGAP*, GASE, and rampIoD are 2.29×10^{-2}, 4.54×10^{-2}, and 6.76×10^{-2}, respectively. Since our approach *liteGAP* requires a less number of messages to be exchanged in the network, less energy is consumed for wireless communication by *liteGAP*.

Second, the run time is measured by changing the number of protocol executions, and the results are presented in Fig. 7. As shown in Fig. 7, the run time of *liteGAP*, GASE, and rampIoD become greater as the number of executions is increased from 10 times to 100 times. Since the protocols are executed repeatedly, a longer run time will be required to run the protocols more times. As a result, the overall run time of three schemes will obviously experience an increment as the number of executions increases. For rampIoD, it is always the most time-consuming protocol when the number of executions is varied from 10 to 100 times. This is because rampIoD is implemented based on heavy-weight techniques such as authenticated encryption with associative data and elliptic curve cryptography. It is widely known that elliptic curve point multiplication is an expensive operation. As a result, a longer time can be expected certainly when rampIoD is executed. And when we change the number of executions, the run time of rampIoD increases significantly. GASE is a group authentication protocol designed for edge computing environments, where secret sharing scheme and aggregated message authentication code are adopted to achieve the group authentication. Compared to the techniques used in rampIoD, secret sharing scheme and aggregated message authentication code are regarded as light-weight operations, and they will take less time to execute. Thus, GASE finishes the authentication process more quickly, and a smaller run time is observed by GASE when the number of executions is increased. Our approach *liteGAP* shows the lowest run time compared to rampIoD and GASE with a varying number of executions. Instead of executing the same authentication scheme for each drone in the group, *liteGAP* is able to realize the group authentication between a drone swarm and the ZSP. Thus, the lowest execution time is observed by *liteGAP*. Note that rampIoD has to execute the entire authentication scheme for each drone in the group so that they can achieve the group authentication. Even through GASE is a group authentication protocol, however, it is not designed for IoD systems and secret sharing scheme is more time-consuming than the techniques used in *liteGAP*.

Third, we measure the run time of *liteGAP*, GASE, and rampIoD by changing the number of drones in the network in Fig. 8. Overall, the increasing number of drones in the network will make the run time of all three schemes increase. The rationale is that certain operations will be executed more times when the number of drones is increased. Finally, a longer run time is observed for all three schemes. *liteGAP* and GASE show a lower run time than rampIoD because rampIoD is a one-to-one authentication protocol, not a group authentication protocol. When the network has more drones, it is obvious that rampIoD will take more time to authenticate them because each drone will need a separate authentication. The run time of *liteGAP* is lower than that of GASE because the run time of secret sharing scheme in GASE significantly increases when the number of drones increases in the network.

Fourth, the CPU time of *liteGAP*, GASE, and rampIoD are measured against the number of executions and the number of drones in Fig. 9 and Fig. 10, respectively. Unlike the

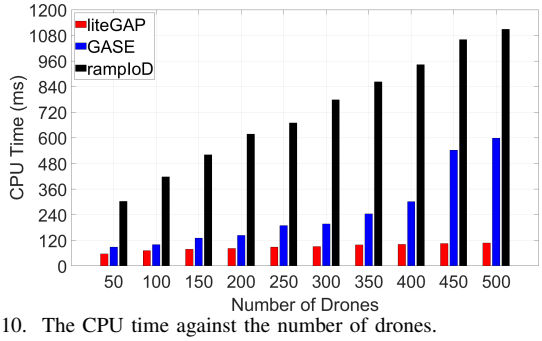


Fig. 10. The CPU time against the number of drones.

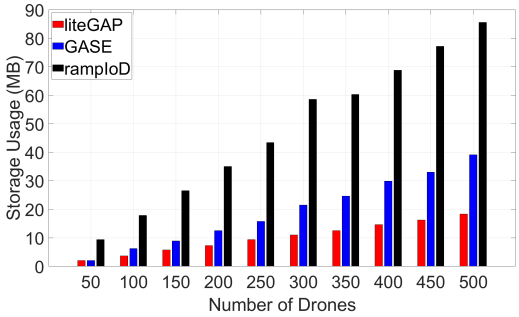


Fig. 11. The storage usage against the number of drones.

run time, the CPU time does not include the latency due to operating in low-power idle state as well as input/output operation delay. As shown in Fig. 9, an increasing number of executions results in an increment in CPU time for all three schemes. However, our scheme *liteGAP* still demonstrates the lowest CPU time because of the adoption of lightweight operations such as bitwise XOR, PUF, and hash function. In addition, the group authentication between multiple drones and the ZSP can be achieved by our scheme *liteGAP*, thus, the number of authentication operations is reduced and a lower CPU time is obtained. A lower CPU time is also obtained by GASE compared to rampIoD, because GASE is designed for group authentication with light-weight techniques (i.e., secret sharing scheme and aggregated message authentication code). The highest CPU time belongs to rampIoD because it adopts resource-hungry techniques such as authenticated encryption with associative data and elliptic curve cryptography. Fig. 10 also shows that our approach *liteGAP* provides the lowest CPU time as the number of drones is increased from 50 to 500.

Fifth, we obtain the storage usage of *liteGAP*, GASE, and rampIoD by changing the number of drones, and present the results in Fig. 11. rampIoD requires the largest amount of memory storage to run. This is because the authenticated encryption with associative data and elliptic curve cryptography are more complex than the techniques being used in both *liteGAP* and GASE. As a result, more space would be needed for instructions, environmental stack, as well as data by rampIoD. Compared to secret sharing scheme and aggregated message authentication code, bitwise XOR, PUF, and hash function do not have high storage demand. Thus, a lower storage usage is obtained by our approach *liteGAP*.

Finally, in Table IV we measure and present the results of average storage usage for *liteGAP*, GASE, and rampIoD. Overall, the average storage usage of our approach *liteGAP* is

TABLE IV
COMPARISON OF AVERAGE STORAGE USAGE

Protocol	No. of Drones	Avg. Storage Usage
liteGAP	200	7.34 MB
GASE	200	12.50 MB
rampIoD	200	35.00 MB

much lower than that of GASE and rampIoD. When the IoD system comprises 200 drones, the storage space required by *liteGAP* is approximately 7.34 MB. However, for GASE and rampIoD, 12.5 MB and 35 MB storage space are consumed, respectively. The reason behind this interesting result is that our approach *liteGAP* adopts lightweight operations which execute faster and use less storage space. GASE and rampIoD use more complex operations such as secret sharing scheme and authenticated encryption with associative data scheme, respectively. Thus, more storage space is consumed by them.

VIII. CONCLUDING REMARKS AND FUTURE WORK

In this paper, we focused on the issue of information privacy and security in the IoD environment, and then designed *liteGAP*, a lightweight group authentication protocol, for IoD systems. With *liteGAP*, a drone swarm and the ZSP can realize the authenticated key establishment concurrently. We first implemented *liteGAP* in HLPSL and performed a security verification on AVISPA, and *liteGAP* is believed to be secure and there is no security weaknesses. Moreover, we conducted a formal and informal security assessment on *liteGAP*, showing that *liteGAP* meets all the pre-defined security requirements. Finally, we built an experimental simulation framework, implemented *liteGAP* and its counterparts in Java, and then evaluated and analyzed their performance. The experimental results showed that *liteGAP* can provide better performance than the state-of-the-art schemes.

Although *liteGAP* outperforms existing schemes, we still see potential for further improvements. To be specific, *liteGAP* does not support cross-domain group authentication that the drones authenticate with the ZSPs located in different physical domains. Nonetheless, how to realize the process of authentication and key agreement between a group of drones and different ZSPs in the IoD environment is a non-trivial problem. Recently, some researchers adopt blockchain technique to resolve the issue of cross-domain authentication. Unfortunately, these blockchain-based security protocols require the frequent update of cryptographic information stored in the blockchain, which incurs a very high communication and computation overhead. As a future work, we plan to look into this potential problem, and propose a lightweight cross-domain group authentication protocol for the IoD systems.

REFERENCES

[1] *Commercial Drone Market*, Last accessed: April 25, 2022, https://www.grandviewresearch.com/.

[2] P. Boccadoro, D. Striccoli, and L. Grieco, "An extensive survey on the Internet of Drones," *Ad Hoc Networks*, vol. 122, p. 102600, 2021.

[3] S. Hussain, S. Chaudhry, O. Alomari, M. Alsharif, M. Khan, and N. Kumar, "Amassing the Security: An ECC-Based Authentication Scheme for Internet of Drones," *IEEE Systems Journal*, vol. 15, no. 3, pp. 4431–4438, 2021.

[4] Y. Wu, H. Dai, H. Wang, and K. Choo, "Blockchain-Based Privacy Preservation for 5G-Enabled Drone Communications," *IEEE Network*, vol. 35, no. 1, pp. 50–56, 2021.

[5] X. Yan and M. Ma, "A Privacy-Preserving Handover Authentication Protocol for a Group of MTC Devices in 5G Networks," *Elsevier Computers & Security*, vol. 116, p. 102601, 2022.

[6] *SPAN*, Last accessed: June 29, 2023, http://people.irisa.fr/Thomas.Genet/span/.

[7] M. Nakkar, R. AlTawy, and A. Youssef, "GASE: A Lightweight Group Authentication Scheme With Key Agreement for Edge Computing Applications," *IEEE Internet of Things Journal*, vol. 10, no. 1, pp. 840–854, 2023.

[8] M. Tanveer, A. Khan, N. Kumar, and M. Hassan, "RAMP-IoD: A Robust Authenticated Key Management Protocol for the Internet of Drones," *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 1339–1353, 2022.

[9] C. Feng, B. Liu, Z. Guo, K. Yu, Z. Qin, and K. Choo, "Blockchain-based Cross-domain Authentication for Intelligent 5G-enabled Internet of Drones," *IEEE Internet of Things Journal*, vol. 9, no. 8, pp. 6224–6238, 2022.

[10] G. Li and C. Lai, "Platoon Handover Authentication in 5G-V2X," in *Proc. IEEE CNS*, 2020, pp. 1–2.

[11] M. Abdel-Malek, K. Akkaya, A. Bhuyan, and A. Ibrahim, "A Proxy Signature-Based Drone Authentication in 5G D2D Networks," in *Proc. IEEE VTC-Spring*, 2021, pp. 1–7.

[12] A. Yazdinejadna, R. Parizi, A. Dehghantanha, and H. Karimipour, "Federated learning for drone authentication," *Elsevier Ad Hoc Networks*, vol. 120, p. 102574, 2021.

[13] S. Zhang and J. Lee, "A Group Signature and Authentication Scheme for Blockchain-Based Mobile-Edge Computing," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4557–4565, 2020.

[14] Z. Shang, M. Ma, and X. Li, "A Secure Group-Oriented Device-to-Device Authentication Protocol for 5G Wireless Networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 11, pp. 7021–7032, 2020.

[15] A. Lizardo, R. Barbosa, S. Neves, J. Correia, and F. Araujo, "End-to-end secure group communication for the Internet of Things," *Journal of Information Security and Applications*, vol. 58, p. 102772, 2021.

[16] A. Yang, D. Boshoff, Q. Hu, G. Hancke, X. Luo, J. Weng, K. Mayes, and K. Markantonakis, "Privacy-Preserving Group Authentication for RFID Tags Using Bit-Collision Patterns," *IEEE Internet of Things Journal*, vol. 8, no. 14, pp. 11 607–11 620, 2021.

[17] A. Miglani and N. Kumar, "Blockchain-Based Co-Operative Caching for Secure Content Delivery in CCN-Enabled V2G Networks," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 4, pp. 5274–5289, 2023.

[18] A. Miglani, N. Kumar, A. Kishore, and N. Mohammad, "UAV-enabled Edge Computing and Blockchain based Secure Charging Station Selection for Energy Trading in V2G Environment," in *Proc. ACM DroneCom*, 2022, pp. 103–108.

[19] M. Mispan and B. Halak, *Physical Unclonable Function: A Hardware Fingerprinting Solution*. Springer, 2021.

[20] J. Wallrabenstein, "Practical and Secure IoT Device Authentication Using Physical Unclonable Functions," in *Proc. IEEE FiCloud*, 2016, pp. 99–106.

[21] F. MacWilliams and N. Sloane, *The Theory of Error Correcting Codes*. Elsevier, 1977.

[22] Q. Do, B. Martini, and K. Choo, "The role of the adversary model in applied security research," *Elsevier Computers & Security*, vol. 81, pp. 156–181, 2019.

[23] W. Chen, Y. Dong, and Z. Duan, "Manipulating Drone Position Control," in *Proc. IEEE CNS*, 2019, pp. 1–9.

[24] W. Stallings, *Cryptography and Network Security - Principles and Practices, 8th Edition*. Pearson, 2020.

[25] D. M'Raihi, S. Machani, M. Pei, and J. Rydell, "TOTP: Time-Based One-Time Password Algorithm," *Internet Request for Comments*, 2011.

[26] G. Jayaraman and V. Dhulipala, "FEECS: Fuzzy-Based Energy-Efficient Cluster Head Selection Algorithm for Lifetime Enhancement of Wireless Sensor Networks," *Arabian Journal for Science and Engineering*, pp. 1–11, 2021.

[27] J. Katz and A. Lindell, "Aggregate Message Authentication Codes," in *Topics in Cryptology*, 2008, pp. 155–169.

[28] *Automated Validation of Internet Security Protocols and Applications*, Last accessed: June 29, 2023, https://www.ercim.eu/publication/Ercim_News/enw64/armando.html.

[29] *VirtualBox*, Last accessed: June 29, 2023, https://www.virtualbox.org/.

[30] W. Mao and C. Boyd, "Towards Formal Analysis of Security Protocols," in *Proc. IEEE CSFW*, 1993, pp. 147–158.

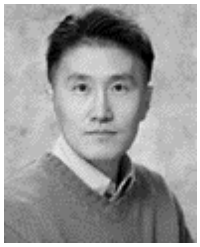
[31] *Eclipse*, Last accessed: June 29, 2023, https://www.eclipse.org/downloads/.

[32] C. Pu and S. Lim, "A Light-Weight Countermeasure to Forwarding Misbehavior in Wireless Sensor Networks: Design, Analysis, and Evaluation," *IEEE Systems Journal*, vol. 12, no. 1, pp. 834–842, 2018.

Cong Pu is an Assistant Professor in the Department of Computer Science, Oklahoma State University, Stillwater, Oklahoma, United States. His primary research interests include network security, data privacy, applied cryptography, wireless networking, and mobile computing.

Clare Warner is an undergraduate student with the Department of Computer Science, Oklahoma State University, Stillwater, Oklahoma, United States.

Kim-Kwang Raymond Choo is a Professor and holding the Cloud Technology Endowed Professorship in the Department of Information Systems and Cyber Security, The University of Texas at San Antonio, San Antonio, TX, United States. His primary research interests include blockchain, big data analytics, cybersecurity and digital forensics.



Sunho Lim is an Assistant Professor in the Department of Computer Science, Texas Tech University, Lubbock, TX, United States. His primary research interests include cybersecurity, mobile data management and privacy, aerial computing, and wireless networks and mobile computing.

Imtiaz Ahmed is an Assistant Professor in the Department of Electrical Engineering and Computer Science, Howard University, Washington, DC, United States. His primary research interests include wireless communications, signal processing, and computer networks.