Privacy-Preserving and Fault-Tolerant Data Aggregation Protocol for Internet of Drones

Cong Pu
Oklahoma State University, United States. Email: cong.pu@ieee.org

***Abstract*—As drones are becoming an indispensable part of society, the idea of Internet of Drones (IoD) has became a commercial reality. In the IoD framework, drones are dispatched to incessantly collect target information in the task region, and intermittently report the observations to a nearby ground station. Then, the ground station summarizes and combines drone observations to reduce the amount of data transmission before sending them to the data requester. As for IoD applications, however, the drones' observational data usually include target's and/or nearby entity's private information. Thus, the privacy of target and/or nearby entity might be accidentally disclosed at the ground station during the data aggregation process. To address the privacy leakage issue, this paper proposes a privacy-preserving and fault-tolerant data aggregation protocol (hereafter referred to as *PriTAP*) for IoD systems. In the *PriTAP*, after receiving all drones' observational data, the ground station first checks and detects the corrupted observational data due to bad wireless channels, and then decrypts the sum of all valid observational data. During the process of data aggregation, however, the ground station cannot access any individual drone's observational data. We perform security verification in the AVISPA environment, where the *PriTAP* has been proved to be safe and reliable in the adversarial setting. We also conduct extensive performance evaluation to validate the performance of *PriTAP*. Experimental results demonstrate that the *PriTAP* not only provides low computational cost, but also efficiently detects corrupted observational data.**

***Index Terms*—Internet of Drones, Privacy Leakage, Data Aggregation, Secure, Privacy-Preserving, Fault Tolerance**

I. Introduction

With significant technological advances in lithium-ion battery technology, ultra-dense microchip, and carbon fiber composites, drones become increasingly affordable and applicable in diverse civilian and commercial settings [1]. According to the research report from ''Drone Industry Insights'' [2], the commercial drone market is forecasted to be worth an approximate $56 billion in 2030. In a short time, various stunning applications of drone technology such as drone light show, public space observation and guidance, and so on have spread rapidly around the globe. For example, during the 2022 FIFA World Cup, the night sky of Doha has been lit up by 80 LED-mounted drones, seamlessly morphing into several World Cup-inspired visuals. As the Internet of Things (IoT) technology continues to mature, there have been tremendous efforts to replace stationary ''things'' with mobile ''drones'' in the recent past. Consistent efforts have successfully produced Internet of Drones (IoD) [3], a revolutionary aerial-ground communication framework.

979-8-3503-8211-2/23/$31.00 ©2023 IEEE

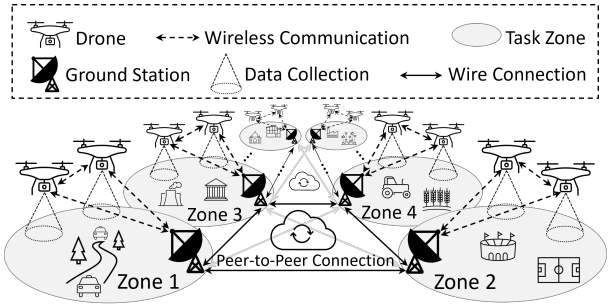

Fig. 1. IoD architecture and example applications. Zone 1: traffic surveillance; Zone 2: sport & entertainment; Zone 3: industrial plants monitoring; and Zone 4: precision agriculture.

In the IoD framework, drones, IoT devices, as well as communication infrastructures interconnect through various types of connection in a way that enables effective information gathering, sharing, and processing. To be specific, as shown in Fig. 1, the IoD framework virtually partitions airspace (or geographical area) into task zones. In each task zone, there is one or multiple ground stations which communicate with nearby drones for task-specific operations (e.g., retrieving traffic information or collecting data from ground IoT devices). With the prevalence of advanced wireless connectivity technologies, it is anticipated that the vast array of valuable IoD applications (e.g., insurance claim, precision agriculture, etc.) will emerge and become a reality [4]. Compared to its ancestor, vehicular networks, where the movement of vehicles is constrained by the road network, the drones in the IoD systems are endowed with greater freedom of movement (mandatory prerequisite: comply with the relevant rules/laws). In addition, the drones' activity arena is the airspace. Thus, the roadway safety can be improved as the vehicular traffic is transferred to the sky (i.e., Pfizer uses drones to deliver COVID-19 vaccines in African countries [5]).

Although there are apparent benefits of such a revolutionary framework, some challenges deserve engineers' full attention and scientific input from academic researchers. The IoD is a generic architecture where security and privacy are not built-in properties but added on as an afterthought. Thus, a plentiful of security and privacy attacks attempt to exploit this design flaw against the IoD systems and achieve the adversarial objectives [6]. Taking drone-assisted autonomous driving as an example, drones are deployed to take images and videos which are used to detect far-away objects for autonomous driving vehicles to operate safely [7]. However, the images and videos captured by drones might be misused to identify, locate, and track pedestrians, which violates the privacy of pedestrians.

Over the last couple of years, several authentication protocols [8]–[11] have been studied to secure the data exchange between an individual drone and the ground station in the IoD systems. However, these schemes either do not perform data aggregation at the ground station or adopt complex operations to fulfill the security and privacy requirements. Data aggregation currently is still an under-explored area in the IoD domain, however, a few data aggregation schemes [12]–[14] have been investigated in vehicular networks. These schemes apply message compression and/or duplicated data reduction techniques to realize the reduction of communication overhead, but they are incapable of achieving fault tolerance during the process of data aggregation. In addition, these schemes are not privacy preserving because the individual vehicle-collected data has to be accessible/visible for the operations of data aggregation. Thus, what has been missing in the IoD community is a data aggregation protocol that adopts resource-friendly computing operations to achieve privacy-preserving data aggregation with the detection capability of corrupted drone observation. The realization of such an approach would be unprecedented because the similar technique is not currently available in the IoD community, and the proposed work will fill this research gap.

Inspired by the above discussion, in this paper we propose a privacy-preserving and fault-tolerant data aggregation protocol (hereafter referred to as *PriTAP*) for IoD systems. The basic idea of *PriTAP* is that the ground station takes advantage of private stream aggregation mechanism [15] to decrypt the sum of all drones' observational data, but is unable to regain any individual drone's observational data. In addition, the *PriTAP* endows the ground station with the ability to detect the corrupted drone observation due to bad wireless channels and aggregate all valid observational data. To verify the safety of *PriTAP*, we choose the High-Level Protocol Specification Language (HLPSL) [16] to implement *PriTAP* and perform a safety verification in the Automated Validation of Internet Security Protocols and Applications (AVISPA) environment [17]. The outputs of AVISPA has proved that the *PriTAP* is a safe and reliable security protocol under adversarial conditions. We also conduct extensive performance evaluation to validate the performance of *PriTAP*. Extensive experimental results demonstrate that the *PriTAP* not only provides low computational cost, but also efficiently detects corrupted observational data.

The rest of the paper is organized as follows. We review the existing literature in Section II. System and adversarial models, and the protocol objectives are provided in Section III. In Section IV, we present the data aggregation protocol. Section V demonstrates the process of security verification and analysis. We conduct an experimental study and analyze the results in Section VI. Finally, we conclude the paper in Section VII.

II. Related Work

Even though data aggregation has not been explored in the IoD domain, some work has been done in other environments. The authors in [18] use drones as aerial base stations to aggregate the collected data from ground IoT devices when the service of existing communication infrastructure is not available. However, their research mainly focus on the trajectory optimization of drones, rather than the data aggregation technique. In [19], the authors study the scheduling issue of data collection/aggregation in wireless sensor networks, where a latency-optimized scheduling scheme is proposed to collect different types of data. They also take into account data collision when aggregating the sensory data, however, the privacy of sensory data is not protected by the approach. An encrypted data aggregation mechanism is proposed for smart grid network in [20], where the gateway router groups the encrypted energy consumption data. In [21], the authors propose a privacy-preserving data aggregation scheme for mobile crowdsensing environment, where the additive secret sharing technique is adopted to protect data privacy. The above two [20], [21] are promising approaches to realize the data privacy, but they fail to guarantee the device's identity privacy.

The researchers in [22] design an anonymous aggregation authentication protocol for safety early warning system in vehicular networks. The protocol is able to aggregate vehicles' signcrypted warning messages into an aggregated ciphertext and restore warning messages all together. In [23], the encrypted data from sensor nodes in the IoT networks are aggregated by the aggregators before sending to the server. Nevertheless, if the message/data is corrupted, the protocol in [22], [23] is unable to detect the error and recover the rest of correct messages/data. The authors in [24] focus on data aggregation in the maritime transportation system, where the maritime sensors are deployed to collect and aggregate the local marine information. Then, the collected data are encrypted and finally transmitted to the server. However, the major drawbacks of the above approach is that the data privacy and the error detection are not considered at all.

Over the past several years, authentication and key agreement protocols have been investigated for IoD systems. In [25], the authors adopt federated learning technique to train deep neural network model with the radio frequency of drones and achieve mutual authentication between the ground station and the drones. The advantage of using federated learning is that it is unnecessary to synchronize the system setting between drones and the ground station. The authors [26] design a group authentication protocol for drone networks, where the new drone is verified by the group leading drone before it can join the drone network and communicate with other drones. In [27], a delegation based authentication scheme is proposed for device-to-device networks, where the drone uses its proxy signature to authenticate itself with other drones in the network. The authors in [28] develop a handover authentication mechanism so that the performance of handover process can be improved when the vehicular platoon changes the contact point of aerial networks in space-air-ground integrated vehicular networks. However, all the abovementioned studies fail to suggest the security solution through which a group of encrypted data can be securely and privately aggregated.

To sum up, what has been lacking in the current theory is a

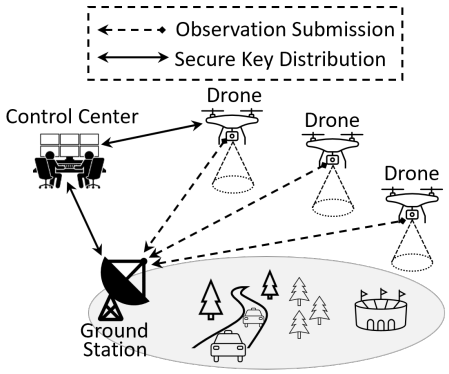

Fig. 2. System model.

data aggregation protocol that can aggregate drones' encrypted data with resource-friendly and privacy-preserving computing operations as well as detect any corrupted data due to bad wireless channels. The major contribution of the proposed research is that it is the first privacy-preserving and fault-tolerant data aggregation protocol for IoD systems, and will suggest a new research direction for the IoD community.

III. System and Adversarial Models & Objectives of The Protocol

A. System Model

The system model is portrayed in Fig. 2, where there are three major participants: control center, ground station, as well as drones. The control center is a trusted participant; it registers each drone through exchanging critical information for the calculation of drone's private key and pseudonym. After enrollment, a batch of drones are dispatched to collect target information in the task region, and periodically report the encrypted observational data to a nearby ground station. In order to avoid storing secret information (e.g., private key) in the memory directly, the integrated circuits of drones are produced with physical unclonable functions (PUF) primitive [29], and the secret information can be restored via PUF when needed. Since drones might be operating in a rugged environment, the encrypted observational data is highly likely to be corrupted due to bad wireless channels. After receiving the encrypted observational data from drones, the ground station will decrypt the sum of all uncorrupted observational data and transmit them to the control center over the secure channel. In this paper, the ground station is also considered as a trusted participant.

B. Adversarial Model

The well-known Dolev-Yao threat model [30] is considered in the system. According to the Dolev-Yao threat model, the external adversary aims to eavesdrop the wireless communication between the drones and the ground station to access drones' observational data. In addition, the external adversary is able to capture the drone using special equipment and attempts to maliciously compromise the drone. However, this malicious attempt will inevitably change or even destroy the PUF, as a result, an invalid PUF-based secret information (e.g., the pseudonym of drone) will be generated. Therefore, we assume that the interval adversary (i.e., compromised drone) does not exist in the system. In summary, the goal of the external adversary is to obtain access to drones' observational data as well as disrupt the data aggregation operation. The external adversary might launch other cyber attacks, e.g., flooding attack [31], however, they are outside the scope of this paper.

TABLE I
Notations

Notation	Meaning
$\mathbb{G}$	Cyclic additive group
P	An arbitrary generator of $\mathbb{G}$
q	Large prime order of $\mathbb{G}$
$H_a(\cdot)$	Secure hash function, H_a:$\{0,1\}^* \rightarrow \mathbb{G}$
$H_b(\cdot)$	Secure hash function, H_b:$\{0,1\}^* \rightarrow \mathbb{Z}_q^*$
D_i	Drone D_i
RID_i	Drone D_i's real identity
che_i	Drone D_i's PUF challenge
$F_{puf}(\cdot)$	PUF
res_i	Drone D_i's PUF response
r_i	Random number generated by drone D_i
PID_i	Drone D_i's pseudonym
$H(\cdot)$	Secure hash function, H:$\{0,1\}^m$ and $m \in \mathbb{Z}$
s_i	Random number generated by control center for D_i
n	The number of registered drones
$\mathbb{PA}^k$	The k^{th} drone pairing
PA_{xy}^k	Drone pair $\{D_x, D_y\}$ in $\mathbb{PA}^k$
PR_x	Drone D_x's private key
PR_{xy}^k	PA_{xy}^k pairing key in $\mathbb{PA}^k$
t_j	The j^{th} periodic interval
$d_i[t_j]$	Drone D_i's observational data in t_j
$c_i^{t_j}$	Ciphertext of $d_i[t_j]$
$D[t_j]$	The sum of all drones' observational data in t_j

C. Objectives of The Protocol

We identify the following security and performance objectives to be met by the proposed protocol: (i) Confidentiality: The drone's observational data is unintelligible to the external adversary; (ii) Anonymity: The drone uses the pseudonym, rather than the real identity, for the communication with the ground station; (iii) Privacy Guarantee: The individual drone's observational data is not visible to the ground station; (iv) Data Aggregation: The ground station can decrypt the sum of all uncorrupted drones' observational data; (v) Fault Tolerance: The ground station can detect the corrupted drone observation and still aggregate all uncorrupted drone observations; and (vi) Computational Complexity: The computational complexity of the proposed protocol is lower than existing schemes.

IV. The Proposed Data Aggregation Protocol

In this section, we describe the proposed privacy-preserving and fault-tolerant data aggregation protocol, which we refer to as *PriTAP* in the following. In the *PriTAP*, the drones collect target information, encrypt the observational data with their private keys, and report the encrypted data to the ground station. Taking advantage of private stream aggregation mechanism [15], the ground station is able to decrypt the sum of all drones' observational data, but does not have access to

any individual drone's observational data. If the ground station fails to obtain the aggregated observational data, it will execute the error detection algorithm to detect the corrupted drone observation, and then aggregate all uncorrupted observational data. In summary, the *PriTAP* is composed of six phases: (i) initialization; (ii) drone registration; (iii) arbitrary pairing; (iv) data encryption; (v) data aggregation; and (vi) error detection. Table I lists all notations used in this paper.

A. Initialization Phase

In this phase, the control center initializes public system parameters and functions in the following steps:

1) The control center chooses a cyclic group $\mathbb{G}$ of the large prime order q with an arbitrary generator P.
2) The control center chooses two secure hash functions H_a and H_b, where H_a:$\{0,1\}^* \rightarrow \mathbb{G}$ and H_b:$\{0,1\}^* \rightarrow \mathbb{Z}_q^*$.
3) The control center advertises all public system parameters and functions as $\{\mathbb{G}, q, P, H_a, H_b\}$.

B. Drone Registration Phase

In this phase, the control center registers the drone D_i in the following steps:

1) The drone D_i chooses its real identity RID_i and PUF challenge che_i.
2) The drone D_i feeds che_i into its PUF $F_{puf}(\cdot)$ to compute the corresponding PUF response $res_i = F_{puf}(che_i)$.
3) The drone D_i generates a random number r_i and calculates its pseudonym $PID_i = H(RID_i \parallel res_i \parallel r_i)$, where H:$\{0,1\}^m$ is a set of fixed length (saying m bits) strings.
4) The drone D_i shares $(RID_i, PID_i, res_i, r_i)$ with the control center via a secure channel.
5) The control center generates and shares a random number s_i with the drone D_i via a secure channel.
6) The drone D_i stores $\{RID_i, che_i, r_i, s_i\}$ in the memory.
7) The control center stores $\{RID_i, PID_i, res_i, r_i, s_i\}$.

C. Arbitrary Pairing Phase

In this phase, the control center arbitrarily pairs all registered drones φ times. Here, we assume that the number of registered drones is an even number *n*. A dummy drone is added if the actual number of registered drones *n* is an odd number. In the k^{th} ($1 \leq k \leq \varphi$) drone pairing $\mathbb{PA}^k$, the underlying steps will be executed:

1) The control center arbitrarily selects two drones, D_x and D_y, and makes a pair $PA_{xy}^k = \{D_x, D_y\}$.
2) The control center calculates the private key PR_x and PR_y for the drone D_x and the drone D_y respectively, $PR_x = H_b(res_x \parallel r_x \parallel s_x)$ and $PR_y = H_b(res_y \parallel r_y \parallel s_y)$.
3) The control center generates the pairing key $PR_{xy}^k \in \mathbb{Z}_q^*$ so that $PR_{xy}^k + PR_x + PR_y = 0 \bmod q$, and shares PR_{xy}^k with the ground station via a secure channel.
4) The control center performs the above three steps for all $\frac{n}{2}$ pairs, where each pairing key PR_{xy}^k is associated with $PA_{xy}^k \in \mathbb{PA}^k$. For the k^{th} drone pairing $\mathbb{PA}^k$, it is easy to obtain $PR^k + \sum_{x=1}^{n} PR_x = 0 \bmod q$, where $PR^k = \sum_{PA_{xy}^k \in \mathbb{PA}^k} PR_{xy}^k \bmod q$.
5) The control center shares all φ drone pairings $\mathbb{PA}^1, \mathbb{PA}^2, \mathbb{PA}^3, \cdots, \mathbb{PA}^{\varphi}$ with the ground station via a secure channel. Note that any drone pair $\{D_x, D_y\}$ should be different in all φ drone pairings.

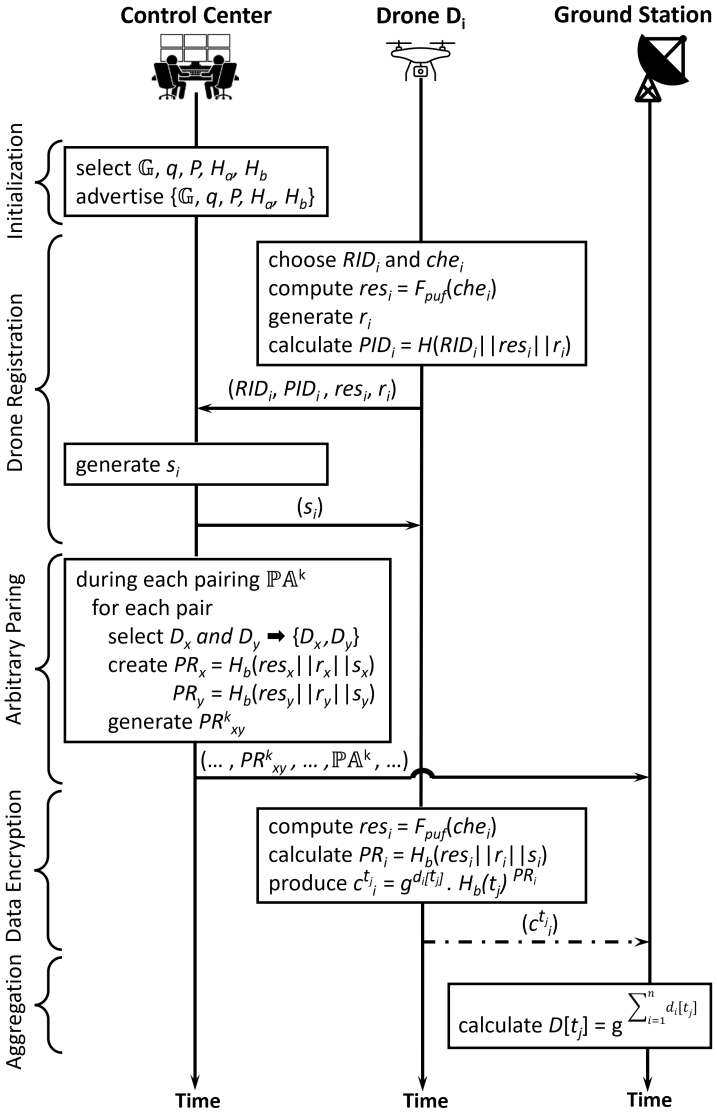

Fig. 3. Privacy-preserving and fault-tolerant data aggregation protocol, where solid arrow lines indicate secure channel and dash-dotted line represents insecure channel.

D. Data Encryption Phase

In this phase, the drone D_i reports its observational data $d_i[t_j]$ to the ground station in the j^{th} periodic interval t_j in the following steps:

1) The drone D_i calculates the PUF response $res_i = F_{puf}(che_i)$ with the PUF challenge che_i.
2) The drone D_i computes the private key $PR_i = H_b(res_i \parallel r_i \parallel s_i)$.
3) The drone D_i encrypts the observational data $d_i[t_j]$ with PR_i to produce the ciphertext $c_i^{t_j} = g^{d_i[t_j]} \cdot H_a(t_j)^{PR_i}$.

4) The drone D_i reports $c_i^{t_j}$ to the ground station via an insecure wireless channel.

E. Data Aggregation Phase

In this phase, the ground station decrypts the sum of all drones' observational data $D[t_j]$ received during the j^{th} periodic interval t_j as follows:

$$\begin{aligned} D[t_j] &= \prod_{i=1}^{n} c_i^{t_j} \cdot H_a(t_j)^{PR^k} \\ &= g^{\sum_{i=1}^{n} d_i[t_j]} \cdot H_a(t_j)^{PR^k + \sum_{i=1}^{n} PR_i} \\ &= g^{\sum_{i=1}^{n} d_i[t_j]} \cdot H_a(t_j)^0 \\ &= g^{\sum_{i=1}^{n} d_i[t_j]}. \end{aligned}$$

Here, $1 \leq k \leq \varphi$, $PR^k = \sum_{PA_{xy}^k \in \mathbb{PA}^k} PR_{xy}^k \bmod q$, and $PR^k + \sum_{i=1}^{n} PR_i = 0 \bmod q$. The value of $\sum_{i=1}^{n} d_i[t_j]$ can be easily obtained through the brute-force search [15]. Since drones might be operating in a rugged environment, the ciphertext $c_i^{t_j}$ is highly likely to be corrupted due to bad wireless channels. As a result, the ground station is unable to obtain the sum $\sum_{i=1}^{n} d_i[t_j]$ because $\sum_{i=1}^{n} d_i[t_j]$ cannot be separated from $\prod_{i=1}^{n} c_i^{t_j} \cdot H_a(t_j)^{PR^k}$. In that case, the ground station will execute the error detection algorithm to detect the corrupted observational data.

F. Error Detection Phase

The basic idea of error detection algorithm is to use φ drone pairings $\mathbb{PA}^1$, $\mathbb{PA}^2$, $\mathbb{PA}^3$ $\cdots$, $\mathbb{PA}^\varphi$ to detect the corrupted observational data. First, for each drone pair $PA_{xy}^1 = \{D_x, D_y\}$ in the 1^{th} drone pairing $\mathbb{PA}^1$, the ground station tries to decrypt the sum of their observational data $d_x[t_j]$ and $d_y[t_j]$ with the ciphertext $c_x^{t_j}$ and $c_y^{t_j}$ as well as the pairing key PR_{xy}^1 according to the following,

$$\begin{aligned} d_{x,y}[t_j] &= c_x^{t_j} \cdot c_y^{t_j} \cdot H_a(t_j)^{PR_{xy}^1} \\ &= g^{d_x[t_j]+d_y[t_j]} \cdot H_a(t_j)^{PR_{xy}^1 + PR_x + PR_y} \\ &= g^{d_x[t_j]+d_y[t_j]} \cdot H_a(t_j)^0 \\ &= g^{d_x[t_j]+d_y[t_j]}. \end{aligned}$$

Here, the value of $(d_x[t_j] + d_y[t_j])$ can be easily obtained through the brute-force search [15]. If the ground station is able to obtain the result of $(d_x[t_j] + d_y[t_j])$, the ciphertext $c_x^{t_j}$ and $c_y^{t_j}$ are believed to be valid. Otherwise, either $c_x^{t_j}$ or $c_y^{t_j}$, or both of them are corrupted. After verifying all $\frac{n}{2}$ drone pairs in the 1^{th} drone pairing $\mathbb{PA}^1$, the ground station can filter out all corrupted drone pairs (e.g., assume to be w pairs, where $1 \leq w \leq \frac{n}{2}$). Second, for all w invalid drone pairs in $\mathbb{PA}^1$, the ground station verifies each suspected drone's observational data in the pair with the remaining φ - 1 drone pairings $\mathbb{PA}^2$, $\mathbb{PA}^3$, $\cdots$, $\mathbb{PA}^\varphi$. If the ground station cannot decrypt the suspected drone's ciphertext on all φ - 1 drone pairings, the suspected drone's ciphertext is believed to be corrupted. Otherwise, the suspected drone's ciphertext is valid. Third, for all valid drones' ciphertexts, the ground station decrypts the sum of their observational data. The major operations of error detection algorithm are summarized in Algorithm 1.

Algorithm 1: Error Detection Algorithm

Input: $\mathbb{PA}^1$, $\mathbb{PA}^2$, $\mathbb{PA}^3$ $\cdots$, $\mathbb{PA}^\varphi$, C^{t_j}

/* $\mathbb{PA}^i$: the i^{th} drone pairing */
/* φ: the total number of drone pairings */
/* C^{t_j}: the set of encrypted observations */
/* IC^{t_j}: the set of invalid encrypted observations */
/* t_j: timestamp */
Function DetectInvalidCipherPair($\mathbb{PA}^1$, C^{t_j}):
  for $i \leftarrow 1$ **to** $\frac{n}{2}$ *by* 1, $w \leftarrow 0$, $IC^{t_j} \leftarrow \emptyset$ **do**
    /* the i^{th} drone pair PA_{xy}^1 in the 1^{th} drone pairing $\mathbb{PA}^1$ */
    if $(d_x[t_j] + d_y[t_j])$ *is obtainable* **then**
      /* the drone D_x's and D_y's ciphertext is $c_x^{t_j}$ and $c_y^{t_j}$ */
      both $c_x^{t_j}$ and $c_y^{t_j}$ are valid;
    else
      /* either $c_x^{t_j}$ or $c_y^{t_j}$, or both are invalid */
      $IC^{t_j} \leftarrow IC^{t_j} \cup c_x^{t_j} \cup c_y^{t_j}$;
    end
  end
Function DetectInvalidCipher(IC^{t_j}, w, $\mathbb{PA}^2$, $\cdots$, $\mathbb{PA}^\varphi$):
  /* for each drone whose cipher is in IC^{t_j} */
  for $i \leftarrow 1$ **to** $|IC^{t_j}|$ *by* 1 **do**
    flag $\leftarrow$ **true**;
    /* the remaining φ - 1 drone pairings, $\mathbb{PA}^2$, $\mathbb{PA}^3$, $\cdots$, $\mathbb{PA}^\varphi$ */
    for $k \leftarrow 2$ **to** φ *by* 1 **do**
      /* $d_x[t_j]$ belongs to a drone in the new pair in the $\mathbb{PA}^k$ */
      if $(d_i[t_j] + d_x[t_j])$ *is obtainable* **then**
        $c_i^{t_j}$ is valid;
        $IC^{t_j} \leftarrow IC^{t_j} - c_i^{t_j}$;
        flag $\leftarrow$ **false**;
        break;
      else
        continue;
      end
    end
    /* $d_i[t_j]$ and any other drone's data are not obtainable in all φ - 1 drone pairings */
    if *flag is* **true then**
      $c_i^{t_j}$ is corrupted;
    end
  end
Function AggregateValidCipher(C^{t_j}, IC^{t_j}):
  decrypt the sum of all valid ciphertexts in (C^{t_j} - IC^{t_j});

V. Security Verification

If the security protocol has potential design flaws or vulnerabilities, it might become a target in the cyber attacks (e.g., masquerading attacks or replay attacks) where the adversary attempts to compromise the objectives of security protocol. In order to prove that the *PriTAP* does not have any design flaw or vulnerability, AVISPA [17] is chose for security verification. Here, AVISPA is a specific security protocol and application verification tool that can automatically analyze the behaviors of protocol and application and validate their security features. Moreover, AVISPA will also verify whether the protocol and application can function securely even under worst-case adversarial environments. AVISPA provides two

```
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL
PROTOCOL
  /home/span/testsuite/results/PriTAP.if
GOAL
  As Specified
BACKEND
  CL-AtSe
STATISTICS
  Analysed: 484 states
  Reachable: 330 states
  Translation: 0.06 seconds
  Computation: 1.20 seconds
```

(a)

```
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/testsuite/results/PriTAP.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 7.24s
  visitedNodes: 876
  nodes  depth: 6 plies
```

(b)

Fig. 4. Security verification results using AVISPA's CL-AtSe and OFMC.

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. The first step of conducting security verification in AVISPA is to implement the security protocol and application in HLPSL [32] which is a AVISPA-specific programming language. After that, AVISPA will evaluate the security protocol and application under masquerading attacks, replay attacks, and other unknown attacks. If the security protocol and application are vulnerable to a specific attack, AVISPA will output a sequence diagram showing the vulnerable scenario. Otherwise, the security protocol and application are marked as ''safe'' by AVISPA. The experiments are conducted in Virtual Box [33], where we set up and configure a fully-functional SPAN+AVISPA [32] environment. The outputs of OFMC and CL-AtSe are shown in Fig. 4, where we can easily observe that the *PriTAP* is identified as a safe security protocol. Meanwhile, we also can conclude that the *PriTAP* does not have any design flaws or vulnerabilities that could be exploited by masquerading attacks, replay attacks, and other unknown attacks.

VI. Performance Evaluation

In this paper, we build a simulation-based experimental environment within Eclipse [34] and conduct extensive experiments to evaluate the performance of *PriTAP*. Specifically, we install Eclipse on a Windows desktop computer, implement *PriTAP* and benchmark schemes in Java programming language. The desktop computer runs Windows 10 Pro 64-bit operating system with the 4th Generation Intel(R) Core(TM) i5-4690K CPU (6M Cache, up to 3.90 GHz). For performance comparison and analysis, we select two benchmark schemes, *SATS* [23] and *SETCAP* [10]. The basic idea of *SATS* is that the IoT devices send the ciphertext of observational data to the aggregation node. After receiving all ciphertexts from IoT devices, the aggregation node simply concatenates all ciphertexts and forwards the aggregated message to the server. Finally, the server separately decrypts each aggregated message and restores the observation data from IoT devices. In

TABLE II
Communication Overhead

Scheme	No. of Transmitted Msg	Communication Energy Cost
*SATS**	48	5.465597×10^{-3}
SETCAP$^{\diamond}$	120	13.66399×10^{-3}
PriTAP‡	40	4.554664×10^{-3}

*: The *SATS* requires each IoT device to send one (1) message piggybacked with the encrypted data to the aggregation node and each aggregation node to forward one (1) aggregated message to the server. Here we assume that there are eight (8) aggregation nodes.
$^{\diamond}$: The *SETCAP* exchanges two (2) messages to negotiate a secret session key between the drone and the ground station, and uses one (1) message to submit the observational data.
‡: The *PriTAP* only needs one (1) message transmitted from the drone to the ground for the submission of encrypted observational data. The messages sent in the phase of drone registration and arbitrary pairing are not counted because they are transmitted via secure channel (e.g., physical medium). Note that the messages exchanged during the preparation phase of *SATS* and *SETCAP* are not counted either.

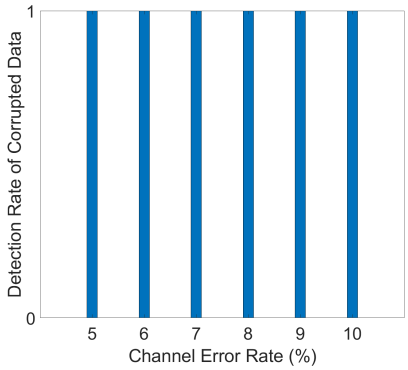

Fig. 5. The performance of detection rate of corrupted data against the channel error rate.

the *SETCAP*, the ground station first shares credentials with each drone, and then assigns them a set of distinct data to collect. Before transmitting data, the drone will authenticate with the ground station and establish a data type specific session key through message concatenation, hash function, and XOR operation. Since the *SETCAP* does not originally support data aggregation, we implement them to decrypt the encrypted data, and then aggregate plaintext data.

In terms of performance metrics, we consider communication overhead, execution time as well as CPU time. Since the simulation-based experiments are conducted on a single machine and actual wireless communication between different entities are not simulated, thus, the communication overhead is represented as the number of transmitted messages and the energy consumption of message transmissions [35]. Moreover, the execution time is expressed as the amount of elapsed time from when the scheme begins running to when the scheme finishes running. Finally, the CPU time is the amount of time for which the CPU was busy with the operations of algorithms. For our scheme *PriTAP*, we also obtain the detection rate of corrupted data.

First, we measure the communication overhead in terms of the number of transmitted messages and the communication energy cost for *SATS*, *SETCAP*, as well as *PriTAP* in Table. II. In this experiment, we assume forty (40) drones are dispatched to collect target information in the task region, and report

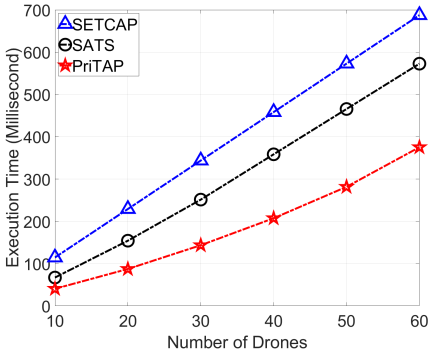

Fig. 6. The performance of execution time against the number of drones.

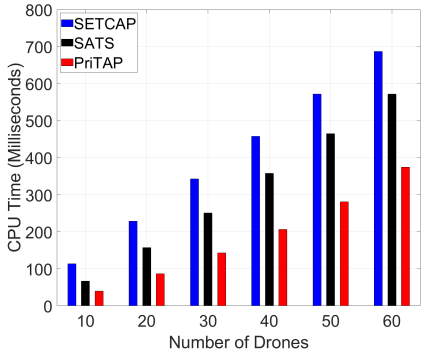

Fig. 7. The performance of CPU time against the number of drones.

the observations to either the ground station or the user. In the *SATS*, the IoT devices send one (1) message to the aggregation node. And then, the aggregation node forwards the aggregated message to the server. Thus, a total of 48 messages are transmitted by 40 drones to submit their observational data to the server. As for the *SETCAP*, it exchanges two (2) messages to negotiate a secret session key between the drone and the ground station, and uses one (1) message to submit the observational data. In such a way, 40 drones will need a total of 120 messages to submit their observational data to the ground station. For our scheme *PriTAP*, only 40 messages are required for a group of 40 drones to submit their observational data to the ground station over an insecure channel. Specifically, in the data encryption phase, the drone encrypts the observational data with its private key and sends it to the ground station. Since each drone only requires one message to submit the observational data, 40 drones will transmit 40 messages to the ground station. We also calculate the communication energy cost for *SATS*, *SETCAP*, as well as *PriTAP*. The *SETCAP* consumes the largest amount of energy because it requires the largest number of messages to submit all drones' observational data. The *SATS* has a higher communication energy cost than our scheme *PriTAP* because it transmits more messages. Our approach *PriTAP* only consumes 4.554664×10^{-3} (joule) because the smallest number of messages are generated and transmitted.

Second, we measure the detection rate of corrupted observational data with varying channel error rate in Fig. 5. Since drones might be operating in a rugged environment, the encrypted observational data is highly likely to be corrupted due to bad wireless channels. In order to detect the corrupted observational data, we design an error detection algorithm where the ground station will filter out all corrupted drone pairs with the 1^{th} drone pairing. After that, the ground station verifies each drone in all suspected drone pairs with the remaining drone pairings. Since the corrupted observational data cannot be decrypted with any other encrypted observational data, it can be successfully detected after verifying all drone pairings. A shown in Fig. 5, the detection rate of corrupted observational data can be 100% in the *PriTAP*, which is consistent with our theoretical analysis result.

Third, we measure the execution time of three schemes by changing the number of drones in Fig. 6. In this experiment, we assume that no encrypted observational data is corrupted during the transmission. Since the *SETCAP* only creates a secret session key between the communication entities, we choose Blowfish [36] symmetric cipher to encrypt the observational data which will be used for data aggregation. For the *SATS*, we also let IoT devices encrypt the observational data with Blowfish symmetric cipher. Overall, the execution time of *SATS*, *SETCAP*, and *PriTAP* increase as the number of drones increases. As for the *SETCAP*, each drone requires a secret session key to encrypt the observational data. When the ground station receives the encrypted observational data, it needs to perform decryption to retrieve the original observational data. Thus, when the number of drones increases, the number of authentications, session key establishments, as well as observational data encryption and decryption will increase correspondingly. As a result, the execution time of *SETCAP* increases. Since the *SATS* performs data aggregation on the encrypted observational data, thus, it observes a smaller execution time than the *SETCAP*. However, compared to our scheme *PriTAP*, the *SATS* still provides higher execution time because of frequently executing Blowfish encryption algorithm. The execution time of *PriTAP* is lower than that of *SATS* and *SETCAP*. This is because the *PriTAP* verifies the identity of drone when the ground station performs the data aggregation. If the drone is not the authenticated entity with the pre-established secret key, its encrypted observational data cannot be decrypted. Since there is no dedicated authentication and key establishment process, a less amount of execution time is observed. In addition, the *PriTAP* will perform the data aggregation with all encrypted observational data, rather than decrypting each encrypted observational data and combining them together. Thus, the *PriTAP* outperforms *SATS* and *SETCAP* in terms of execution time. When the number of drones increases, the *PriTAP* will need more time to perform data aggregation on more encrypted observational data. An increasing execution time is observed for the *PriTAP* as the number of drones increases.

Finally, we measure the CPU time with varying number of drones in Fig. 7. In this experiment, we assume that no encrypted observational data is corrupted during the transmission. As the number of drones increases, the CPU time of

SATS, *SETCAP*, and *PriTAP* also increase accordingly. This is because more drones will cause more authentications and submit more encrypted observational data to the user or the ground station. Thus, the user or the ground station has to perform more decryptions and aggregate more observational data, as a result, more CPU time will be required by each scheme. However, the lowest CPU time still belongs to the *PriTAP*. In the *PriTAP*, the ground station will aggregate all encrypted observational data together, instead of decrypting and combining each observational data. Thus, a lower CPU time is obtained by the *PriTAP*.

VII. Conclusion

In this paper, we proposed a privacy-preserving and fault-tolerant data aggregation protocol (also called *PriTAP*) for IoD systems, where the communication security and data privacy are being addressed simultaneously. The basic idea of *PriTAP* is that the ground station first detects the corrupted observational data due to bad wireless channels after receiving all drones' observational data, and then decrypts the sum of all valid observational data. During the process of data aggregation, however, the ground station cannot access any individual drone's observational data. To evaluate its security performance, the *PriTAP* was first implemented in the security-sensitive protocol modeling language and evaluated using the AVISPA framework. Finally, we implemented the *PriTAP* and two benchmark schemes, and conducted experimental simulation to evaluate their performance. Our experimental results indicate that the *PriTAP* not only provides superior computational cost performance, but also efficiently detects corrupted observational data.

References

[1] C. Pu and P. Zhu, "Mitigating Routing Misbehavior in the Internet of Drones Environment," in *Proc. IEEE VTC*, 2022-Spring, pp. 1–6.

[2] *7 Drone Stocks to Watch for 2023*, Last accessed: Dec 23, 2022, https://money.usnews.com/investing.

[3] C. Pu, A. Wall, and K. Choo, "Bilinear Pairing and PUF Based Lightweight Authentication Protocol for IoD Environment," in *Proc. IEEE MASS*, 2022, pp. 115–121.

[4] L. Bine, A. Boukerche, L. Ruiz, and A. Loureiro, "Leveraging Urban Computing with the Internet of Drones," *IEEE Internet of Things Magazine*, vol. 5, no. 1, pp. 160–165, 2022.

[5] A. Verma, P. Bhattacharya, D. Saraswat, S. Tanwar, N. Kumar, and R. Sharma, "SanJeeVni: Secure UAV-envisioned Massive Vaccine Distribution for COVID-19 Underlying 6G Network," *IEEE Sensors Journal*, vol. 23, no. 2, pp. 955–968, 2022.

[6] C. Pu, A. Wall, I. Ahmed, and K. Choo, "SecureIoD: A Secure Data Collection and Storage Mechanism for Internet of Drones," in *Proc. IEEE MDM*, 2022, pp. 83–92.

[7] J. Shin, M. Piran, H. Song, and H. Moon, "UAV-Assisted and Deep Learning-driven Object Detection and Tracking for Autonomous Driving," in *Proc. ACM Mobicom - DroneCom Workshop*, 2022, pp. 7–12.

[8] C. Pu, A. Wall, K. Choo, I. Ahmed, and S. Lim, "A Lightweight and Privacy-Preserving Mutual Authentication and Key Agreement Protocol for Internet of Drones Environment," *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9918–9933, 2022.

[9] S. Yu, A. Das, Y. Park, and P. Lorenz, "SLAP-IoD: Secure and Lightweight Authentication Protocol Using Physical Unclonable Functions for Internet of Drones in Smart City Environments," *IEEE Transactions on Vehicular Technology*, pp. 1–15, 2022.

[10] M. El-Zawawy, A. Brighente, and M. Conti, "SETCAP: Service-Based Energy-Efficient Temporal Credential Authentication Protocol for Internet of Drones," *Computer Networks*, vol. 206, p. 108804, 2022.

[11] C. Pu, "A Featherweight Authentication and Key Agreement Scheme for Internet of Drones Applications," in *Proc. IEEE PIMRC*, 2023, pp. 1–6.

[12] B. Bhabani and J. Mahapatro, "CluRMA: A cluster-based RSU-enabled message aggregation scheme for vehicular ad hoc networks," *Vehicular Communications*, vol. 39, p. 100564, 2023.

[13] M. Gillani, H. Niaz, A. Ullah, M. Farooq, and S. Rehman, "Traffic Aware Data Gathering Protocol for VANETs," *IEEE Access*, vol. 10, pp. 23 438–23 449, 2022.

[14] H. Zheng, M. Luo, Y. Zhang, C. Peng, and Q. Feng, "A Security-Enhanced Pairing-Free Certificateless Aggregate Signature for Vehicular Ad-Hoc Networks," *IEEE Systems Journal*, vol. 17, no. 3, pp. 3822–3833, 2023.

[15] R. Shi, R. Chow, and T. Chan, "Privacy-Preserving Aggregation of Time-Series Data," in *US Patent EP2485430A2*, 2011.

[16] Y. Chevalier, L. Compagna, J. Cuellar, P. Drielsma, J. Mantovani, S. Mödersheim, and L. Vigneron, "A High Level Protocol Specification Language for Industrial Security-Sensitive Protocols," in *Proc. SAPS*, 2004, pp. 1–13.

[17] *Automated Validation of Internet Security Protocols and Applications*, Last accessed: Dec 25, 2022, http://www.avispa-project.org.

[18] A. Bera, S. Misra, C. Chatterjee, and S. Mao, "CEDAN: Cost-Effective Data Aggregation for UAV-Enabled IoT Networks," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2022.

[19] V. Pham, T. Nguyen, B. Liu, M. Thai, B. Dumba, and T. Lin, "Minimizing Latency for Data Aggregation in Wireless Sensor Networks: An Algorithm Approach," *ACM Transactions on Sensor Networks (TOSN)*, vol. 18, no. 3, pp. 1–21, 2022.

[20] X. Zhang, W. Tang, D. Gu, Y. Zhang, J. Xue, and X. Wang, "Lightweight Multidimensional Encrypted Data Aggregation Scheme With Fault Tolerance for Fog-Assisted Smart Grids," *IEEE Systems Journal*, vol. 16, no. 4, pp. 6647–6657, 2022.

[21] X. Yan, B. Zeng, and X. Zhang, "Privacy-Preserving and Customization-Supported Data Aggregation in Mobile Crowdsensing," *IEEE Internet of Things Journal*, vol. 9, no. 20, pp. 19 868–19 880, 2022.

[22] Y. Yang, L. Zhang, Y. Zhao, K. Choo, and Y. Zhang, "Privacy-Preserving Aggregation-Authentication Scheme for Safety Warning System in Fog-Cloud Based VANET," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 317–331, 2022.

[23] G. Said, A. Ghani, A. Ullah, M. Azeem, M. Bilal, and K. Kwak, "Light-Weight Secure Aggregated Data Sharing in IoT-Enabled Wireless Sensor Networks," *IEEE Access*, vol. 10, pp. 33 571–33 585, 2022.

[24] C. Wang, J. Shen, P. Vijayakumar, and B. Gupta, "Attribute-based secure data aggregation for isolated IoT-enabled maritime transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2021.

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

[26] Y. Aydin, G. Kurt, E. Ozdemir, and H. Yanikomeroglu, "Group Authentication for Drone Swarms," in *Proc. IEEE WiSEE*, 2021, pp. 72–77.

[27] M. Abdel-Malek, K. Akkaya, A. Bhuyan, and A. Ibrahim, "A Proxy Signature-Based Drone Authentication in 5G D2D Networks," in *Proc. IEEE VTC2021-Spring*, 2021, pp. 1–7.

[28] C. Lai and Z. Chen, "Group-based Handover Authentication for Space-Air-Ground Integrated Vehicular Networks," in *Proc. IEEE ICC*, 2021, pp. 1–6.

[29] G. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proc. ACM/IEEE DAC*, 2007, pp. 9–14.

[30] D. Dolev and A. Yao, "On the Security of Public Key Protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.

[31] C. Pu and P. Zhu, "Defending against Flooding Attacks in the Internet of Drones Environment," in *Proc. IEEE GLOBECOM*, 2021, pp. 1–6.

[32] *SPAN*, Last accessed: Jan 25, 2023, http://people.irisa.fr/Thomas.Genet/span/.

[33] *VirtualBox*, Last accessed: Jan 25, 2023, https://www.virtualbox.org/.

[34] *Eclipse*, https://www.eclipse.org/downloads/.

[35] 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.

[36] B. Schneier, "Description of a new variable-length key, 64-bit block cipher (Blowfish)," in *Proc. International Workshop on Fast Software Encryption*, 2005, pp. 191–204.