IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 28, 2020, accepted September 23, 2020, date of publication October 1, 2020, date of current version October 13, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3028063

Cooperative Privacy-Preserving Data Collection Protocol Based on Delocalized-Record Chains

MERCEDES RODRIGUEZ-GARCIA, MARÍA-ÁNGELES CIFREDO-CHACÓN, AND ÁNGEL QUIRÓS-OLOZÁBAL
Microelectronic Circuit Design Group, Escuela Superior de Ingeniería, Universidad de Cádiz, 11519 Cádiz, Spain
Corresponding author: María-Ángeles Cifredo-Chacón (mangeles.cifredo@uca.es)

This work was supported by the Spanish Government under Project RTC-2016-5095-1 EmPhasys.

ABSTRACT This paper aims to advance the field of data anonymization within the context of Internet of Things (IoT), an environment where data collected may contain sensitive information about users. Specifically, we propose a privacy-preserving data publishing alternative that extends the privacy requirement to the data collection phase. Because our proposal offers privacy-preserving conditions in both the data collecting and publishing, it is suitable for scenarios where a central node collects personal data supplied by a set of devices, typically associated with individuals, without these having to assume trust in the collector. In particular, to limit the risk of individuals' re-identification, the probabilistic k-anonymity property is satisfied during the data collection process and the k-anonymity property is satisfied by the data set derived from the anonymization process. To carry out the anonymous sending of personal data during the collection process, we introduce the *delocalized-record chain*, a new mechanism of anonymous communication aimed at multi-user environments to collaboratively protect information, which by not requiring third-party intermediaries makes it especially suitable for private IoT networks (besides public IoT networks).

INDEX TERMS Anonymous communication, privacy, k-anonymity, Internet of Things, privacy-preserving data collection.

I. INTRODUCTION

Mining personal data from diverse electronic sources, such as wearable devices, electronic health records, online retail services or social networks, is an activity of great interest to public and private entities. Particularly, wearable technology is one of the promising and growing areas of the IoT, allowing tracking and data collecting to enhance health, eldercare, security or lifestyle patterns, with applications in fields as diverse as fitness or monitoring early stage Alzheimer's patients. Personal data collecting and processing allow obtaining statistical conclusions regarding to health, education, employability or consumer habits among others, making this activity crucial to enhance the business decision-making [1], health care [2] or for offering a customized on-line experience [1].

However, when personal data are made available for secondary use, the individuals' privacy may be compromised. In order to overcome ethical and legal issues arising from the processing and transfer of personal data, governments have developed laws to protect the personally identifiable information [3], being some of the most important the Privacy Act (PA) [4] and the Health Insurance Portability and Accountability Act (HIPAA) [5] at the United States of America, and the General Data Protection Regulation (GDPR) [6] at the European Union.

Personally identifiable information includes not only direct identifiers (e.g., social security number or IP address), but also quasi-identifiers, that is, non-identifying personal data that, with an adequate cross-referencing, may lead to find out the individual identity [7], [8]. Nowadays, individuals' re-identification constitutes a real threat against privacy that is being employed by data brokers [1] to build user profiles for commercial and business purposes, or even to carry out discriminatory practices in fields such as health insurance management or personnel selection.

To minimize the likelihood of individuals' re-identification, collected personal data must be subjected to an anonymization process before being shared. Since the main objective of data collection is to conduct statistical analyses, the

The associate editor coordinating the review of this manuscript and approving it for publication was Longxiang Gao.

anonymization process should preserve the data analytical utility as much as possible. This implies that the conclusions or inferences extracted from the anonymous data must be similar to those obtained from the original data. With the goal of balancing privacy and utility preservation, several models and methodologies have been proposed in the scope of Privacy-Preserving Data Publishing (PPDP) [9]–[11], such as k*-anonymity* or *probabilistic* k*-anonymity* privacy models, and *microaggregation* or *generalization* masking methods.

PPDP methodologies mainly focus on the problem of anonymizing large volumes of data collected by a central party. In these cases, the responsibility for the anonymization of the data lies with the organism which collects them, whereas data holders do not play any role in this process. They have to trust the ability and integrity of the collecting entities. This may be an acceptable approach in some cases, such as the anonymization of the data set obtained from the patients of a hospital to be employed in statistical studies by the scientific community, but in other less reliable scenarios would be advisable to preserve privacy throughout the data collection process (i.e., before the data reach the central collector). In this way, each individual generating data could participate in the anonymizing tasks avoiding a possible compromising use of his/her personal information, either by a negligent or a dishonest action from the central collector.

The provision of mechanisms that preserve privacy during the data collection process, would not only be beneficial for the individuals who generate the data, but also for the companies that collect and process them, since the former would be more willing to share their personal information. With these objectives, different protocols have been proposed in the scope of Privacy-Preserving Data Collection (PPDC) [12], [13]. However, these PPDC protocols either have the disadvantage of being vulnerable to network traffic analysis attacks or have the limitation of requiring that a significant part of the communications between the data sources and the data collector be carried out through anonymous channels of third parties, such as Tor or VPN providers, which can be a problem for data collection processes deployed in private networks.

In this paper, we propose a new privacy-preserving method to generate k-anonymous data sets that extends the privacy requirement to the data collection phase, without the data holders having to put their trust in the data collector. Our main contributions are as follows:

1) We define the concept of anonymity in the context of personal data collection in terms of *delocalization* and *unlinkability*.
2) We introduce a novel mechanism of collaborative anonymous communication aimed at multi-user environments, named *delocalized-record chain*. Our proposal is characterized by being an autonomous solution adapted to the distributed nature of an IoT environment, in which the users interested in getting anonymity work in synergy to anonymize their data transmissions. Because our solution lacks third-party intermediaries, it is particularly appropriate for private networks, such as private IoT networks.
3) We propose a new data collection protocol to generate k-anonymous data sets, named *Cooperative Privacy-Preserving Data Collection protocol* (cPPDC), that offers privacy-preserving conditions in both data collection and publication, without limiting PPDP method that can be used to k-anonymize the data set. Our protocol is resistant to network traffic analysis attacks by using the *delocalized-record chain* as a data transmission medium in the collection phase.
4) We present an anonymity analysis of the proposed protocol against network traffic analysis attacks and collusion attacks, and give a comparison with related protocols.

The rest of the paper is organized as follows. Section II provides a background on PPDP, PPDC, and anonymous communication channels. Section III defines the assumptions of our work, and the anonymity requirements that must be fulfilled in the data collecting process. Section IV presents our method of collaborative anonymous communication and Section V describes the cPPDC protocol. Section VI presents an anonymity analysis of the proposed protocol and a comparison with related protocols. Finally, conclusions and future research are summarized in Section VII.

II. BACKGROUND

A. PRIVACY-PRESERVING DATA PUBLISHING (PPDP)

PPDP provides a set of non-cryptographic methods intended to anonymize personal data while preserving the statistical usefulness of the information. Personal information is classified as:

–Direct identifiers: Data that uniquely identify an individual, e.g., passport number, social security number or IP address.

–Quasi-identifiers: Data that, on their own, do not identify an individual, but in combination (e.g., age + ZIP code + occupation) may be used to re-identify individuals by cross-referencing them with additional sources of identifying information via data linking attacks [7]. Any data is potentially a quasi-identifier, depending on the external information available to the attacker [14].

–Confidential attributes: Data that contain sensitive information whose disclosure could result in harm, embarrassment, inconvenience, or unfairness to individuals, e.g., health condition.

To minimize the identity disclosure and, consequently, the possibility of gaining confidential information about a specific individual, the data must be subjected to a de-identification process before being shared. As a result, data collectors publish a modified version of the original data set, where the direct identifiers have been removed and the quasi-identifiers have been altered (or masked) to satisfy certain privacy guarantees. Unlike identifiers, quasi-identifiers must not be removed because they provide valuable information for data analysis.

Different privacy models have been proposed within PPDP to protect quasi-identifiers with provable privacy guarantees while maintaining (part of) their statistical utility. One of the most common privacy models is k-anonymity [8], which is based on homogenizing information to reduce its identifiability. The individuals' records in the data set are homogenized by creating groups of at least k records sharing the same values in their quasi-identifiers. Since each combination of quasi-identifiers in the k-anonymous data set is indistinguishable from at least k-1 other records, the probability of re-identifying an individual from the data set is limited to $1/k$. The k parameter allows for adjusting the balance between data privacy and integrity. The larger the k value, the more anonymous the data set will be. Instead, lower k values result in less alteration of data, thus yielding more useful k-anonymous data sets for analysis. An alternative approach that relaxes the indistinguishability requirement of k-anonymity is probabilistic k-anonymity [14], which only requires that the probability of re-identification be the same as in k-anonymity, i.e., at most $1/k$.

Mainly, two PPDP methods can be used to k-anonymize the quasi-identifiers in a data set:

–Generalization and suppression [15]: The quasi-identifier values are replaced by a range, if they are numerical (e.g., the individual's age is replaced by an age range), or by a class, if they are nominal (e.g., the individual's occupation is replaced by an occupational category). Suppression contributes to reduce the amount of generalization required to generate the k-anonymous data set by removing outlier values. This approach has the disadvantage of requiring a high computational cost to find an optimal generalization that minimizes the information loss while satisfies k-anonymity.

–Microaggregation [16], [17]: This method, more practical than the previous, partitions the data set into groups of at least k records following a criterion of maximum similarity on the quasi-identifiers. Then, the quasi-identifier values in each group are replaced by the group representative value, typically the average value. This method is usually applied to numerical data, although variants for nominal data exist [18], [19].

To maximize statistical utility of the anonymized data set, the quasi-identifiers of all individuals must be previously known to properly define the ranges and classes needed to generalize the quasi-identifier values, or the groups of k records used by microaggregation, always with the target of minimizing the information loss. This fact implies that the k-anonymization process has to be centralized in the collector.

B. PRIVACY-PRESERVING DATA COLLECTION (PPDC)

In an attempt to protect data at source, different strategies have been proposed in the scope of Privacy-Preserving Data Collection (PPDC) [12], [13]. These strategies have in common the segregation of data in the collection process. In a first phase, only the quasi-identifiers are sent to the data collector. With this information, the collector k-anonymizes the set of received quasi-identifiers and distributes the resulting k-anonymous partition to the individuals. In the following phase, the confidential data are sent to the collector along with the k-anonymous group to which the individual belongs. This segregated collection contributes to anonymize data because it allows confidential attributes to be disassociated from original quasi-identifiers. However, all the effort of anonymization at source would be useless if IP addresses of the users' devices are not properly anonymized during the collection process, since this datum is inherently associated to any communication between the individuals and the data collector, and it univocally identifies the data holders.

In order to prevent the disclosure of IP addresses, [13] proposes to use anonymous channels, such as Tor. Besides the vulnerabilities these channels present, they have the disadvantage that they may not be always available for all individuals and collectors.

As an alternative, [12] suggests selecting leaders among the data holders (two leaders by k-anonymous group) to act as intermediaries in the phase of collection of confidential data, thereby preventing direct communications between the individuals and the collector. The individuals of each k-anonymous group send to the first leader their confidential data along with k-1 fake data to prevent real sensitive information disclosure. In parallel, these fake data are also sent to the second leader. Finally, the data collected by the leaders are provided to the collector so that it can obtain the k-anonymized data set by joining records and subtracting fake data. Although it is an interesting proposal, the information could be re-associated by network traffic analysis, since the scheme does not incorporate cryptographic mechanisms to protect data during communication between the parties.

C. ANONYMOUS COMMUNICATION

Different techniques to anonymize the IP addresses of the users' devices in a communication have been proposed in the literature, which we classify as: multi-point intermediary systems and single-point intermediary systems.

The main multi-point intermediary system, known as The Onion Router (Tor) [20], proposes to establish an anonymizing circuit of several forwarding nodes that acts as a multi-point intermediary between the user's device (source node) and the destination node. The purpose of this circuit is to dissociate the payload (i.e., the actual data to be conveyed in the network message) and the destination IP address from the source IP address. For that, both the payload and the destination IP address are concealed by the source node in a nested cryptographic structure named onion, with as many cryptographic layers as forwarding nodes have the Tor circuit, typically entry node, middle node and exit node. As the network message containing the onion passes through the Tor circuit, each forwarding node 1) removes a layer of the onion to discover the IP address of the next hop, and 2) replaces the source and destination IP addresses of the network message by the addresses of the immediately preceding and following nodes. In this circuit, the user's IP address only will be known by the entry node, and the payload and the destination IP address only are known by the exit node. When the message leaves the Tor circuit to be introduced in other networks and reach the destination, it will have as source IP address the address of the exit node, instead of the real source address.

Despite efforts to dissociate the content of the messages from the users' IP addresses, anonymity guarantees of Tor could be compromised in cases where the adversary can statistically correlate network traffic on both ends of the Tor circuit [21]–[26]. For example, if an autonomous system is involved on both ends, it could statistically correlate traffic and infers the destination with which the source node is communicating and, thus, link the content of the message with the identity of the source node [27]–[29].

In single-point intermediary systems, such as systems based on VPN servers [30], [31], a single node (VPN server) acts as an intermediary by accessing to the destination on behalf of the user. A ciphered channel between the user's device (source node) and the VPN server is generated to hide the message from third parties. When the message leaves the VPN server to reach the destination node, it will have as source IP address the address of the VPN server, instead of the real source address. In addition to being more vulnerable to correlation attacks than the multi-point intermediary systems [32], the single-point systems present a second drawback: the VPN server can link the content of the message with the user's IP address, which means that users have to place their trust in VPN providers.

Above techniques achieve anonymity of the source nodes by dissociating the values of certain fields in the network message. In particular, the payload and the destination IP address are dissociated from the source IP address. However, as discussed above, advanced traffic analysis may re-associate this information. In IoT environments, where multiple users have the need to protect their identifying data, it would be desirable to provide a new model of anonymous communication that was multiuser-oriented and autonomous, in which the users interested in getting anonymity work in synergy to protect their information. Thus, instead of introducing uncertainty into the relationship between fields of a network message (inter-field uncertainty), which may be vulnerable to certain traffic analysis, the participating nodes could collaborate to introduce uncertainty inside the payload (intra-field uncertainty) generating an anonymous collective network message.

III. PRELIMINARIES

A. ASSUMPTIONS

We assume that the personal data collection is carried out in an IoT environment, either a private IoT network or a public IoT network, with multiple data generating nodes, called generators, each typically associated with an individual. The generators send data to a data collecting node, called collector, typically associated with an organization.

Generators, represented by G_i, are devices that take part in a data collection process generating and supplying personal data from their owners. In particular, each generator sends a set of quasi-identifiers, Q_i, and a set of confidential attributes, C_i. The generators could be wearable devices designed to gather health and exercise data from individuals, such as biometric data, consumption habits or sleep patterns.

Collector is the party that gathers the personal data supplied by the generators to build a k-anonymous data set that can be shared with third parties while ensuring a minimum level of anonymity to the data holders. The resulting k-anonymous data set is a modified version of the original data set, where the values of the quasi-identifiers have been altered (or masked), Q_i^*, to satisfy privacy guarantees of k-anonymity. A data collector could be a company that collects health and exercise data supplied by the wearable devices from its customers.

We assume participants in the data collection process, i.e., the generators and the collector, are semi-honest. This means that they follow the rules of the data collection protocol, but they may behave maliciously and try to infer personal information about other participants by analyzing the data received during the execution of the protocol.

We assume the collector knows the IP addresses of the generators, and the generators know the IP address of the collector.

B. ANONYMITY IN THE CONTEXT OF DATA COLLECTION

We define the concept of anonymity in the context of personal data collection in terms of *delocalization* and *unlinkability*.

1) *Delocalization requirement*: neither the generators nor the collector can univocally associate the quasi-identifiers or the confidential attributes of a specific individual with the IP address from his/her generator, a piece of information classified as a direct identifier within personally identifiable information.
2) *Unlinkability requirement*: neither the generators nor the collector can univocally associate the original quasi-identifiers of a specific individual with his/her confidential attributes. If this requirement is not met, a malicious generator or collector could use the individual's quasi-identifiers to re-identify him/her through data linking attacks and, consequently, gain confidential information about him/her.

IV. COLLABORATIVE ANONYMOUS SENDING

In this section, we propose a new mechanism of multiuser-oriented, autonomous anonymous communication aimed at IoT environments, where multiple nodes (generators) interested in getting anonymity work in synergy to decouple the data records they wish to send from their IP addresses.

A. DELOCALIZED-RECORD CHAIN

In order to prevent individuals' data records from being associated with the IP addresses of their generators, we propose these be sent jointly to the collector in a collective network message that we call *delocalized-record chain*. We say that a

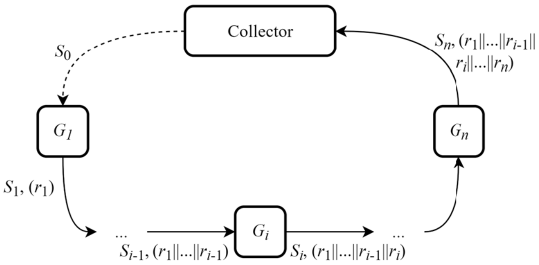

Collector
S_0
G_1
G_n
S_n, $(r_1||...||r_{i\text{-}1}||$
$r_i||...||r_n)$
S_1, (r_1)
...
G_i
...
$S_{i\text{-}1}$, $(r_1||...||r_{i\text{-}1})$
S_i, $(r_1||...||r_{i\text{-}1}||r_i)$

FIGURE 1. Elemental communication procedure.

data record r_i is a *delocalized record* if it cannot be univocally associated with the IP address of its generator G_i.

The procedure to build the delocalized-record chain is represented in Fig. 1. Initially, the collector prepares a list with the IP addresses of all the generators willing to participate in the data collection process, n being the total number of participating generators. Then, the chain is collaboratively built by the generators following the order established in the list. When a generator G_i receives the record chain $(r_1|| \dots ||r_{i-1})$ from its predecessor G_{i-1}, it adds its record r_i to the chain, $(r_1|| \dots ||r_{i-1}||r_i)$, before sending it to the next generator in the list, G_{i+1}. Only when the chain has been completed by the last generator in the list, it is delivered to the collector. As shown in Fig. 1, an additional field, S_i, is transmitted along with the record chain $(r_1|| \dots ||r_{i-1}||r_i)$. S_i is a control information field that allows defining aspects related to the transmission and processing of records, such as the list of IP addresses of the participating generators.

This elementary collaborative submission is the basis for decoupling data records from their IP addresses, but it is far from achieving this objective since the transmitted data are not protected, and it would be easy for any attacker (either an internal adversary, such as a malicious generator or collector, or an external adversary) to associate a particular record with the IP address of its generator simply by following the order of the chain. To reinforce our purpose of delocalization of records during the chain assembly process, we incorporate in our model the protection mechanisms detailed below.

1) DELOCALIZING THE RECORDS IN THE CHAIN

To prevent a particular record from being relocated by its position in the chain, we propose to randomize these positions. Thus, each time a generator has to add its record to the chain, this is placed in a random position among the other records. When the chain is completed, the records will be intermixed, occupying positions different from the order established in the list defined by the collector.

However, since the chain is collaboratively built, any generator can access the content of the chain and view the records added by the previous generators. This causes a circumstance of vulnerability at the beginning of the chain assembly process, when the uncertainty introduced by mixing the records is low (during the addition of the first records) or null (in the case that the added record is the first). To guarantee that the uncertainty introduced by mixing the records be enough to obfuscate their location, we propose that the records can only be revealed when the chain has been completed, i.e., when the chain is delivered to the collector. For that, the collector generates the key pair (PK_c, SK_c), such that the private key SK_c is kept secret by the collector and the public key PK_c is revealed in a previous stage to all generators willing to participate in the process. Then, each participant generator ciphers its record r_i with the public key of the collector before adding it to the chain, the result being denoted by $PK_c(r_i)$. Because only the collector is able to decipher the records with its private key SK_c, we guarantee that the content of the chain will only be revealed when the chain has finished being assembled and delivered to the collector.

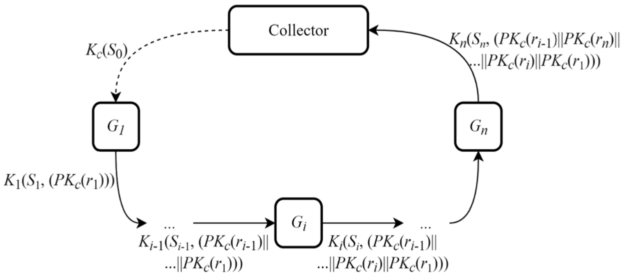

Collector
$K_c(S_0)$
G_1
G_n
$K_n(S_n$, $(PK_c(r_{i\text{-}1})||PK_c(r_n)||$
$...||PK_c(r_i)||PK_c(r_1)))$
$K_1(S_1$, $(PK_c(r_1)))$
...
G_i
...
$K_{i\text{-}1}(S_{i\text{-}1}$, $(PK_c(r_{i\text{-}1})||$
$...||PK_c(r_1)))$
$K_i(S_i$, $(PK_c(r_{i\text{-}1})||$
$...||PK_c(r_i)||PK_c(r_1)))$

FIGURE 2. Delocalized-record chain.

2) PROTECTING THE DELOCALIZED-RECORD CHAIN FROM INCOMING AND OUTGOING TRAFFIC ANALYSIS

By comparing incoming and outgoing traffic from a generator, a malicious collector could identify the ciphered record that the generator has added to the chain and consequently link it with its IP address. To avoid the collector gains useful information from network traffic analysis, the whole chain must be ciphered through symmetric encryption when it is transmitted between generators.

Thus, each time a generator G_i has to send the chain to the next generator G_{i+1}, a session key K_i is shared between them through the Diffie-Hellman method, which enables two parties jointly establish a shared secret key over an insecure channel. Once the session key K_i has been established, G_i ciphers the whole chain with K_i before sending it to G_{i+1}. When the generator G_{i+1} receives the chain, it must decipher it with the same key before adding its record. Fig. 2 represents in detail the conceptual idea of delocalized-record chain.

Since the collector does not know the session keys used during transmission, it will not be able to view the content the chain going in and out of the generators, thus thwarting any attempt to analyze incoming and outgoing traffic. In addition, because each pair of consecutive generators in the chain generates a different session key, no generator will be able to view both the incoming and outgoing chain of another generator, which also prevents malicious generators from succeeding in re-localizing records by analyzing network traffic.

V. COOPERATIVE PRIVACY-PRESERVING DATA COLLECTION PROTOCOL

In this section, we introduce our Cooperative Privacy-Preserving Data Collection protocol (cPPDC), which uses

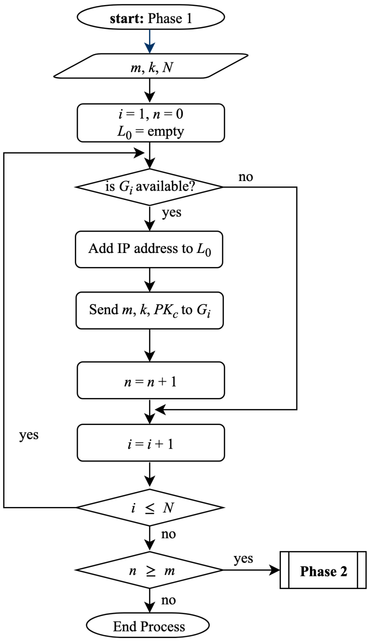


FIGURE 3. Flowchart of the first phase.

the anonymous collaborative sending method proposed in Section IV as the basis of communication and the notion of segregated collection that related works follow.

The cPPDC protocol is composed of four phases:

–In the first phase, the collector prepares the list of generators that will participate in the data collection process.

–In the second phase, the generators send their quasi-identifiers to the collector through the anonymous collaborative sending method proposed in Section IV.

–In the third phase, the collector generates a k-anonymous partition with the collected quasi-identifiers.

–In the fourth phase, using again the method proposed in Section IV, the generators send to the collector their confidential attributes together with the group of the k-anonymous partition in which they have been classified. Finally, the collector generates the k-anonymous data set by joining this information with the masked quasi-identifiers of the k-anonymous partition obtained in the third phase.

A. FIRST PHASE: SETUP

Fig. 3 represents the flowchart of the first phase. The collector starts the setup phase sending an invitation to the generators to participate in the data collection process, N being the number of generators that are candidates to participate in the cPPDC process. Both the value of the minimum number of participants, m, and the minimum number of members of each anonymous group, k, must be previously defined by the collector and sent in the invitation, along with the public key of the collector, PK_c, specified in Subsection IV.A.

Then, the collector creates a list, L_0, with the IP addresses of those generators that accept to participate in the process. Only if the number of generators that accept to participate, n, is equal to or greater than m, the data collection process is started.

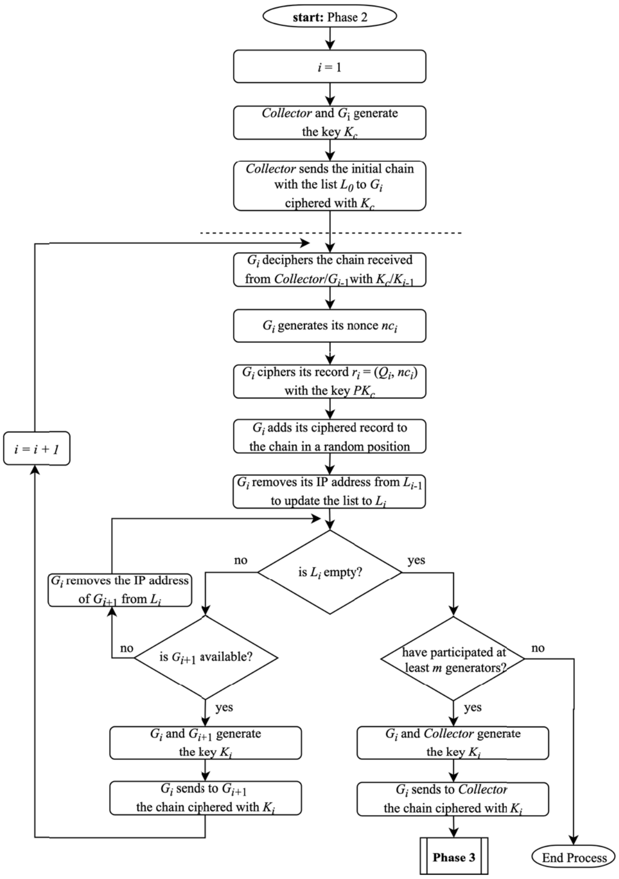

start: Phase 2
$i = 1$
Collector and G_i generate the key K_c
Collector sends the initial chain with the list L_0 to G_i ciphered with K_c
G_i deciphers the chain received from *Collector*/G_{i-1} with K_c/K_{i-1}
G_i generates its nonce nc_i
G_i ciphers its record $r_i = (Q_i, nc_i)$ with the key PK_c
G_i adds its ciphered record to the chain in a random position
G_i removes its IP address from L_{i-1} to update the list to L_i
$i = i + 1$
no
is L_i empty?
yes
G_i removes the IP address of G_{i+1} from L_i
no
is G_{i+1} available?
yes
G_i and G_{i+1} generate the key K_i
G_i sends to G_{i+1} the chain ciphered with K_i
have participated at least m generators?
no
yes
G_i and *Collector* generate the key K_i
G_i sends to *Collector* the chain ciphered with K_i
Phase 3
End Process

FIGURE 4. Flowchart of the second phase.

B. SECOND PHASE: ANONYMOUS SENDING OF QUASI-IDENTIFIERS

In the second phase, the generators collaborate to send anonymously their quasi-identifiers to the collector through the delocalized-record chain defined in Subsection IV.A. The flowchart of this phase is represented in Fig. 4.

The collector starts the process initializing the chain. The initial chain consists only of the initial list of participants, L_0, which is entered into the control information field described in Subsection IV.A., i.e., $S_0 = L_0$. Then, the collector sends the chain to the first generator of the list and transfers control of the process to the generators.

Following the method described in Subsection IV.A., each generator, G_i, adds its ciphered record, $PK_c(r_i)$, to the delocalized-record chain. In this phase, r_i is formed by the set of quasi-identifiers, Q_i, and a nonce [33], nc_i, generated by G_i. The nonces act as pseudonyms of the records and are used by the collector in the next phase to indicate to which group of the k-anonymous partition each record has been assigned. Before sending the chain to the next generator in the list, G_i updates the list of participants deleting its IP address to obtain L_i, such that $S_i = L_i$. When the address list is empty or none of the remaining generators respond, the delocalized-record chain is considered complete. If the number of generators that have finally participated is equal to or greater than m, the chain is delivered to the collector. Otherwise the process is cancelled.

C. THIRD PHASE: K-ANONYMIZATION OF QUASI-IDENTIFIERS

In the third phase, the collector generates a k-anonymous partition $\{P_1, \dots, P_p\}$ with the set of records $r_i = (Q_i, nc_i)$ received in the previous phase. Any PPDP method that satisfies k-anonymity, as those depicted in Section II, can be used to k-anonymize the quasi-identifiers of the set of received records. The resulting partition will be formed by disjoint groups, P_j, of at least k records. Each group P_j contains the nonces of the records belonging to that group, along with the masked quasi-identifiers for that group, i.e., $P_j = \{(nc_{(1)}, \dots, nc_{(s)}), Q_j^*\}$, $nc_{(1)}$ being the first nonce of the group which does not necessarily have to coincide with the nonce nc_1 of the first generator in the chain G_1, $nc_{(s)}$ being the last nonce of the group such that $s \geq k$, and Q_j^* being the masked values of the quasi-identifiers for that group. The masked value for a quasi-identifier is shared by all records of the group and replaces the values that those had originally.

From this partition the collector creates a partition table, T_0, with p rows, one for each partition group. Each row contains the index of the partition group l, the nonces of the records belonging to that group and a counter initialized to 0.

D. FOURTH PHASE: ANONYMOUS SENDING OF CONFIDENTIAL DATA

In the fourth phase, the generators again collaborate to send anonymously their confidential attributes to the collector through a new delocalized-record chain. The flowchart of this phase is represented in Fig. 5.

The collector starts the process similarly to Phase 2, but initializing the chain with $S_0 = (L_0, T_0)$. Again, following the method described in Subsection IV.A, the generators take control of the process to build the delocalized-record chain. First, each generator, G_i, has to locate which group of the k-anonymous partition it belongs to, looking for its nonce in the partition table. Then, G_i creates its data record, r_i, composed by its confidential attributes, C_i, and the index, l, of the group of the k-anonymous partition to which it belongs. Once the record is ciphered with the public key of the collector, $PK_c(C_i, l)$, this is added to the chain. Before sending the chain to the next generator in the list, G_i updates both the list of participants and the partition table. The list is updated in the same way than the second phase, whereas the partition table is updated by increasing in one unit the counter of the corresponding row.

Finally, the last generator in the chain has to check that every counter of the partition table has a value equal to or greater than k. If this requirement is not met, the process must be cancelled. If this phase ends successfully, the collector receives a delocalized-record chain containing the confidential attributes from each generator, along with its partition group index. By joining this information to the k-anonymous partition obtained in the third phase, the collector obtains the k-anonymous data set.

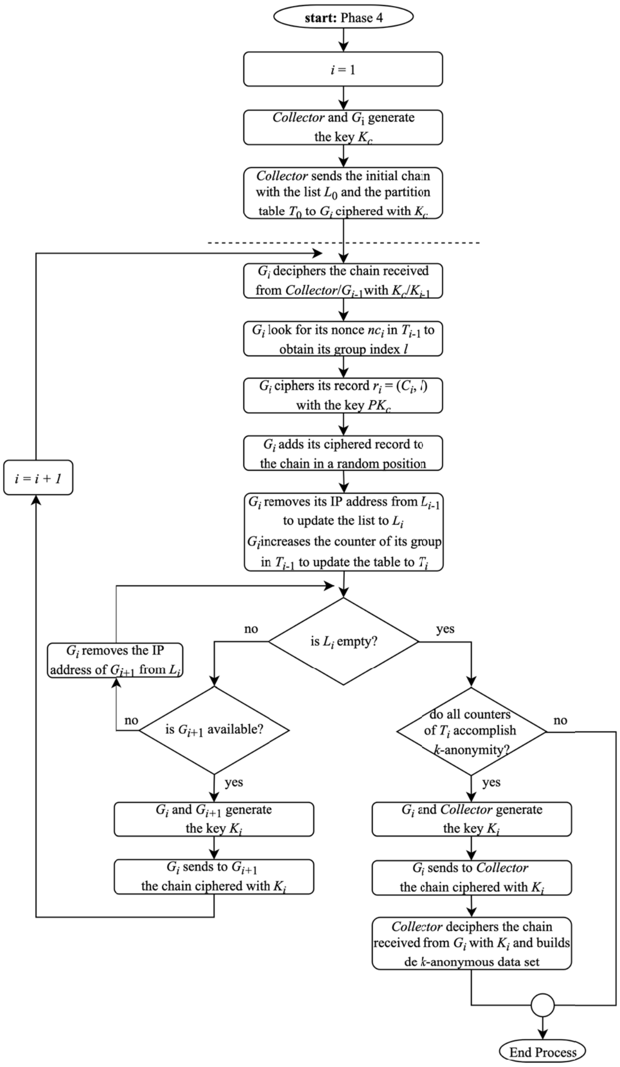

start: Phase 4
$i = 1$
Collector and G_i generate the key K_c
Collector sends the initial chain with the list L_0 and the partition table T_0 to G_i ciphered with K_c
G_i deciphers the chain received from *Collector*/G_{i-1} with K_c/K_{i-1}
G_i look for its nonce nc_i in T_{i-1} to obtain its group index l
G_i ciphers its record $r_i = (C_i, l)$ with the key PK_c
G_i adds its ciphered record to the chain in a random position
G_i removes its IP address from L_{i-1} to update the list to L_i
G_i increases the counter of its group in T_{i-1} to update the table to T_i
$i = i + 1$
no
is L_i empty?
yes
G_i removes the IP address of G_{i+1} from L_i
no
is G_{i+1} available?
yes
do all counters of T_i accomplish k-anonymity?
no
yes
G_i and G_{i+1} generate the key K_i
G_i sends to G_{i+1} the chain ciphered with K_i
G_i and *Collector* generate the key K_i
G_i sends to *Collector* the chain ciphered with K_i
Collector deciphers the chain received from G_i with K_i and builds de k-anonymous data set
End Process

FIGURE 5. Flowchart of the fourth phase.

VI. EVALUATION

In this section, we analyze whether our cPPDC protocol has achieved the anonymity requirements defined in Subsection III.B. We evaluate these requirements in the worst scenario, that is, when the adversary is a participant in the data collection process, since an internal adversary may have more information than an external adversary. We assume that any participant, be it a generator or the collector, is a potential adversary and may therefore be interested in gathering personal data from other participants. In our analysis, we go further and not only evaluate our protocol against isolated adversaries, but also against a collusion of adversaries.

We assume a practical environment where both (incoming and outgoing) network traffic analysis attacks and collusion attacks are possible. By analyzing and correlating network messages or colluding with other participants, the adversaries may gain knowledge about the personal information of their victims. In these terms, we evaluate whether our cPPDC protocol is capable of, on the one hand, preventing adversaries from associating the personal data transmitted during the collection process with the IP addresses of the data holders (*delocalization requirement*) and, on the other hand, preventing adversaries from univocally associating the confidential attributes of a specific individual with his/her original quasi-identifiers (*unlinkability requirement*).

In addition, we analyze the privacy guarantees on the collector side and compare our protocol with other related works in terms of anonymity. Finally, we analyze impact of the cryptographic processing overhead in the data collection phases of our protocol.

A. ANONYMITY ANALYSIS OF THE PROPOSED cPPDC PROTOCOL

Theorem 1: cPPDP satisfies the *delocalization requirement* against (incoming and outgoing) network traffic analysis attacks and collision attacks, unless the collusion were either between the victim's neighbouring generators and the collector or between the second generator and the collector, the victim in this case being the first generator.

Proof:

If the adversary is a generator:

1) Because the individuals' records are encrypted with the public key of the collector, no generator will be able to know their content. Only the collector can know the content by using its private key.
2) Because the individuals' ciphered records are randomly intermixed in the chain, occupying positions different from the order followed in the assembly process, no generator will be able to relocate them by analyzing their position. Only the first ciphered record added to the chain could be relocated by the second generator, although, due to the point 1, the IP address of the relocated record will not be able to be correlated with the content of the record (i.e., with the record in clear text).
3) Because the chain is encrypted when it is transmitted between generators and each pair of consecutive generators uses a session key which is different from those of the other pairs, no generator will be able to examine the incoming and outgoing chains of other generators. A generator can only know the outgoing chain from its direct predecessor. Consequently, no generator will be able to use the comparison of incoming and outgoing traffic to discover the ciphered records that other generators have added to the chain.

Based on points 1-3, it is concluded that a malicious generator cannot univocally associate the quasi-identifiers or the confidential data of a specific individual with his/her IP address, even in cases where incoming and outgoing network traffic analysis attacks were carried out or where the malicious generator were second in the chain.

If the adversary is the collector:

4) Because the collector receives the chain from the last generator, the collector will only be able to associate the set of received records with the IP address of this generator, without being able to determine the IP address of each record.
5) Because the individuals' records are randomly intermixed in the chain (see the point 2), the collector will not be able to relocate them by analyzing their position.
6) Because the chain is transmitted encrypted with session keys unknown to the collector, the collector will not be able to use the comparison of incoming and outgoing network traffic to discover the record that each generator has added to the chain.

Based on points 4-6, it is concluded that a malicious collector cannot univocally associate the quasi-identifiers or the confidential data of a specific individual with his/her IP address, even if the collector carried out an incoming and outgoing network traffic analysis attack.

If the adversary is a collusion of participants:

7) If one generator and the collector were in collusion, due to the points 2 and 3, they could not discover the records that other generators have added to the chain, unless the malicious generator were second in the chain. In the latter case, due to the point 1, the content of the first record could be correlated with the IP address.
8) If the neighbouring generators of a given generator (i.e., the direct predecessor and successor in the chain) were in collusion, due to the point 3, they could discover the record that such generator has added to the chain, but, due to the point 1, they could not know its content.
9) If the neighbouring generators of a given generator and the collector were in collusion, due to the point 3, they could discover the record that such generator has added to the chain, and, due to the point 1, they could also know its content. Consequently, they could correlate the content of the record with the IP address.

Based on points 7-9, it is concluded that a collusion attack cannot succeed in relocating the records, unless the collusion were either between the victim's neighboring generators and the collector or between the second generator and the collector, the victim in this case being the first generator.

Theorem 2: cPPDP satisfies the *unlinkability requirement* against (incoming and outgoing) network traffic analysis attacks and collision attacks, unless the collusion were either between the victim's neighbouring generators and the collector or between the second generator and the collector, the victim in this case being the first generator.

Proof:

If the adversary is a generator:

10) Because all records in the chain are encrypted with the public key of the collector (see the point 1), no generator will be able to view the confidential data of other generators during the fourth phase of the cPPDC protocol or the original values of the quasi-identifiers during the second phase, even if the generator carried out a network traffic analysis.

Based on point 10, it is concluded that a malicious generator cannot univocally associate the confidential data of a specific individual with his/her quasi-identifiers because these data are unknown for it.

If the adversary is the collector, it is necessary to analyze the information that the collector handles during the phases of the cPPDC protocol:

11) During the second phase, the collector obtains $r_i = (Q_i, nc_i)$, i.e., the individuals' original quasi-identifiers along with their pseudonyms.
12) During the third phase, the collector generates a k-anonymous partition with the set of received records. Each k-anonymous group $\{(nc_{(1)}, \ldots, nc_{(s)}), Q_j^*\}$ contains the pseudonyms of the records belonging to that group, $(nc_{(1)},\ldots, nc_{(s)})$, along with the masked quasi-identifiers for that group, Q_j^*. The number of records in each k-anonymous group is greater than or equal to k, i.e., $s \geq k$.
13) During the fourth phase, the collector obtains the individuals' confidential data to associate them with the masked quasi-identifiers, (Q_j^*, C_i).

Based on points 11-13, it is concluded that the collector cannot associate the confidential data C_i of a specific individual with his/her original quasi-identifiers Q_i.

If the adversary is a collusion of participants:

14) If one generator and the collector were in collusion, due to the points 2 and 3, they could not discover the records that other generators have added to the chain, unless the malicious generator were second in the chain. In the latter case, due to the point 1, the content of the first record could be correlated with the IP address, in both phase 2 (during the collecting of quasi-identifiers) and phase 4 (during the collecting of confidential data). Consequently, this collusion could link the confidential data of the first individual in the chain with his/her original quasi-identifiers.
15) If the neighbouring generators of a given generator (victim) were in collusion, due to the point 3, they could discover the record that such generator has added to the chain, but, due to the point 1, they could not know its content. Consequently, they could not link the confidential data of the victim with his/her original quasi-identifiers.
16) If the neighbouring generators of a given generator and the collector were in collusion, due to the point 3, they could discover the record that such generator has added to the chain, and, due to the point 1, they could also know its content. Consequently, they could correlate the content of the record with the IP address, in both phase 2 (during the collecting of quasi-identifiers) and phase 4 (during the collecting of confidential data). Consequently, they could link the confidential data of the victim with his/her original quasi-identifiers.

Based on points 14-16, it is concluded that a collusion attack cannot succeed in data linking, unless the collusion were either between the victim's neighbouring generators and the collector or between the second generator and the collector, the victim in this case being the first generator.

Theorem 3: cPPDP satisfies the *probabilistic k-anonymity* privacy model on the collector side.

Proof:

Based on points 11-13, if the data collector behaved maliciously and tried to infer confidential information about a specific individual by correlating the data received during the execution of the protocol, the collector could, at most, associate the individual's confidential data, C_i,, with the set of original quasi-identifiers of the k-anonymous group to which the individual belongs, $(Q_{(1)}, \ldots, Q_{(s)})$, through the following reverse mapping of the information: $(C_i, Q_j^*) \rightarrow \{Q_j^*, (nc_{(1)}, \ldots, nc_{(s)})\} \rightarrow \{(nc_{(1)}, \ldots, nc_{(s)}), (Q_{(1)}, \ldots, Q_{(s)})\}$.

Therefore, given the confidential data C_i of a specific individual, the collector will be able to determine, at most, the set of k values among which the original quasi-identifiers would be. That is, the probability that the collector correctly correlates C_i and Q_i is at most $1/k$, thereby satisfying the property of probabilistic k-anonymity. Logically, the larger the k value, the greater the uncertainty of the collector will be.

B. COMPARISON WITH OTHER PPDC PROTOCOLS

In this section, we compare our protocol with those PPDC protocols that, like ours, neither limit the PPDP method that can be used to k-anonymize the data set nor require third-party intermediaries to anonymize communications in the data collection process. In particular, our protocol is compared with the PPDC protocol [12] discussed in Subsection II.B, since, as far as we know, it is the only related protocol that meets the above characteristics.

Unlike our protocol, [12] is vulnerable to network traffic analysis attacks because the network messages are transmitted unencrypted. Thus, by capturing and analysing the network messages that the individuals send to the collector in the first phase of the protocol, any adversary could associate the individuals' original quasi-identifiers (conveyed in the payloads of the network messages) with their IP addresses (conveyed in the headers of the network messages), i.e., the adversary could get the trace (IP address, Q_i). In the second phase, the adversary could associate the individuals' confidential data with their IP addresses by capturing and correlate the messages transmitted to the leaders. Specifically, the adversary could get the traces (IP address, Q_j^*, C_i, C_i^{fake}) and (IP address, C_i^{fake}) of the messages addressed to the first leader and the second leader, respectively. By subtracting the second trace from the first one, the adversary would achieve the relation (IP address, C_i). In view of the above, it is concluded that the PPDC protocol [12] does not satisfy the *delocalization requirement*. Consequently, [12] also does not satisfy the *unlinkability requirement*, since any adversary could associate the original quasi-identifiers of the individuals with their confidential attributes by correlating the traces (IP address, Q_i) and (IP address, C_i) by the IP address. Note that both requirements may be infringed by any participant in the protocol through a simple traffic analysis, without even requiring a collusion attack.

Like our protocol, [12] also satisfies the probabilistic k-anonymity privacy model on the collector side. Therefore, the probability that the collector correctly correlates the confidential attributes of a specific individual with his/her original quasi-identifiers is, at most, $1/k$.

C. ANALYSIS OF THE CRYPTOGRAPHIC PROCESSING OVERHEAD

The cryptographic operations carried out by each generator G_i in the data collection are as follows:

1) Secret sharing with the previous generator G_{i-1} in the delocalized-record chain.
2) Symmetric decryption of the incoming chain by using the shared secret as the key.
3) Asymmetric encryption of the record that the generator G_i must add to the delocalized-record chain.
4) Secret sharing with the next generator G_{i+1} in the delocalized-record chain.
5) Symmetric encryption of the outgoing chain by using the new shared secret as the key.

To evaluate the impact of the cryptographic processing overhead in data collection, we considered a simulated network scenario formed by 50 generators with 1800 MHz ARM Cortex A-17 processors of 32 bits. In our simulation, we used strong cryptosystems currently applied in network environments. Specifically, we used RSA-2048 (2048-bit modulus) as the public key cryptosystem, Diffie-Hellman (2048-bit modulus) to share a 256-bit secret, and AES-256 as the symmetric cryptosystem.

Timings of each of the cryptographic operations carried out by the generators in the Phase 2 of our simulation are shown in Table 1, similar results were obtained for Phase 4. Timings are based on the measurements on ARM Cortex A-17 processors reported by the European Network of Excellence for Cryptology [34]. The data records collected in Phase 2 contains one nonce of 128 bits and one quasi-identifier of 344 bits formed by four attributes: *age*, *home city*, *education* and *occupation*. On the collector side, the nominal values of the attributes *education* and *occupation* were associated with concepts modeled in the WordNet ontology [35] to be able to microaggregate nominal data through semantically-grounded PPDP mechanisms [18], [19].

As we can see in Table 1, the processing time for operations 1, 3 and 4 is constant during the data collection, regardless of the position of the generator in the chain. Timings for the encryption/decryption of the chain (operations 2 and 5) depend on the chain size, which changes in each hop of the data collection (each generator adds its encrypted record and removes its IP address from the list). As an example, the incoming chain for G_1 is formed by the list, L_0, with the 32-bit IPv4 addresses of those generators that accept to participate in the process (200 bytes), and the outgoing chain is formed by the updated list, L_1, (196 bytes) and the ciphered record from G_1 (331 bytes). As shown Table 1, secret sharing is the operation with the hardest processing because it includes the calculation of the shared secret key and the generation of the key pair used in the calculation.

TABLE 1. Processing time of the cryptographic operations carried out by a generator in the Phase 2 of the proposed protocol.

Number	Cryptographic operation	Processing time
1	DH secret-sharing with the previous generator	1.244 ms for G_i
2	AES-256 decryption of the incoming chain	0.004 ms for G_1 0.230 ms for G_{50}
3	RSA-2048 encryption of the record	0.079 ms for G_i
4	DH secret-sharing with the next generator	1.244 ms for G_i
5	AES-256 encryption of the outgoing chain	0.008 ms for G_1 0.235 ms for G_{50}

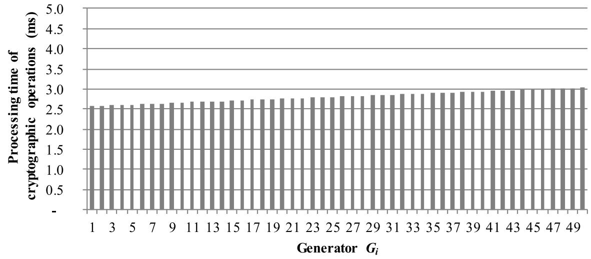


FIGURE 6. Execution time of the cryptographic operations performed by each generator in the Phase 2 of the proposed protocol.

As overview, Fig. 6 shows the time in milliseconds of the set of cryptographic operations performed by each generator in the Phase 2 of the protocol. According to the position of the generators in the chain, the timings range from 2.6 ms to 3.0 ms, which evidences that cryptographic processing overhead does not severely impact on data collection.

VII. CONCLUSION

In this paper, we have presented a new protocol for the privacy-preserving data collection. The proposed protocol is capable of generating k-anonymous data sets in IoT environments, protecting at source the personal data that a set of devices sends to a central collector. To extend the privacy requirement to the data collection phase, our protocol use a new mechanism of collaborative anonymous communication named *delocalized-record chain*. Since our protocol does not require third-party anonymous communication channels, its application is especially relevant in IoT environments deployed on private networks.

As a result, our solution offers privacy-preserving conditions in both the data collection and publication. In particular, our protocol is capable of: (1) preventing adversaries from associating the personal data transmitted during the collection process with the IP addresses of the data holders (*delocalization requirement*), (2) preventing adversaries from univocally associating the confidential attributes transmitted during the collection process with the original quasi-identifiers of their holders (*unlinkability requirement*). Anonymity analysis shows these conditions are fulfilled even in the face of (incoming and outgoing) network traffic analysis attacks and several cases of collusion attacks. Moreover, the probabilistic k-anonymity property is satisfied on the collector side, that is, if the collector tried to infer personal information about the participants by analyzing data received during the execution of the protocol, the probability that the collector would correlate the confidential data of a given individual with their quasi-identifiers is at most $1/k$. Finally, the k-anonymity property is satisfied by the resulting anonymized data set to be published or shared with third parties.

Future work will be devoted to test the benefits of our approach in specific IoT applications, such as IoT ecosystems deployed in private networks with Field-Programmable Gate Arrays (FPGAs) acting as providers of health and exercise data from a community of individuals to a central collector. Finally, we plan to also protect confidential data by incorporating in our scheme other privacy models, such as l-diversity [36] or t-closeness [37].

REFERENCES

[1] E. Ramírez, J. Brill, M. Ohlhausen, J. Wright, and T. Mc-Sweeny, "Data brokers: A call for transparency and accountability," U.S. Federal Trade Commission FTC, Washington, DC, USA, Tech. Rep., May 2014.

[2] M. Elliot, K. Purdam, and D. Smith, "Statistical disclosure control architectures for patient records in biomedical information systems," *J. Biomed. Informat.*, vol. 41, no. 1, pp. 58–64, Feb. 2008.

[3] E. McCallister, T. Grance, and K. Scarfone, "Guide to protecting the confidentiality of personally identifiable information (PII)," U.S. Dept. Commerce, Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. 800-122, 2010.

[4] (1974). *Privacy Act*.[Online]. Available: https://www.justice.gov/opcl/privacy-act-1974

[5] HIPAA. (2004). *Health Insurance Portability and Accountability Act*, [Online]. Available: http://www.hhs.gov/ocr/hipaa/

[6] Regulation (EU) 2016/679 of the European Parliament and of the Council. *Protection of Natural Persons With Regard to the Processing of Personal Data and on The Free Movement of Such Data, and Repealing Directive 95/46/EC (General Data Protection Regulation)*. Accessed: Apr. 27, 2016. [Online]. Available: http://data.europa.eu/eli/reg/2016/679/oj

[7] V. Ciriani, S. Vimercati, S. Foresti, and P. Samarati, "Microdata protection," in *Secure Data Management in Decentralized Systems*. New York, NY, USA: Springer, 2007, pp. 291–321.

[8] L. Sweeney, "K-anonymity: A model for protecting privacy," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 10, no. 05, pp. 557–570, Oct. 2002.

[9] A. Hundepool, J. Domingo-Ferrer, L. Franconi, S. Giessing, E.S. Nordholt, K. Spicer, and P.-P. Wolf, *Statistical Disclosure Control*. Hoboken, NJ, USA: Wiley, 2012.

[10] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments," *ACM Comput. Surveys*, vol. 42, no. 4, pp. 1–53, Jun. 2010.

[11] L. Xu, C. Jiang, J. Wang, J. Yuan, and Y. Ren, "Information security in big data: Privacy and data mining," *IEEE Access*, vol. 2, pp. 1149–1176, 2014.

[12] S. Kim and Y. D. Chung, "An anonymization protocol for continuous and dynamic privacy-preserving data collection," *Future Gener. Comput. Syst.*, vol. 93, pp. 1065–1073, Apr. 2019.

[13] J. Soria-Comas and J. Domingo-Ferrer, "Co-utile collaborative anonymization of microdata," in *Modeling Decisions for Artificial Intelligence* (Lecture Notes in Computer Science), vol. 9321, V. Torra and T. Narukawa Eds. Cham, Switzerland: Springer, 2015, pp. 192–206.

[14] J. Soria-Comas and J. Domingo-Ferrer, "Probabilistic k-anonymity through microaggregation and data swapping," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Jun. 2012, pp. 1–8.

[15] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: K-anonymity and its enforcement through generalization and suppression," Comput. Sci. Lab., SRI Int., Menlo Park, CA, USA, Tech. Rep. SRI-CSL-98-04, 1998.

[16] D. Defays and M. N. Anwar, "Masking microdata using microaggregation," *J. Off. Statist.*, vol. 14, no. 4, pp. 449–461, 1998.

[17] J. Domingo-Ferrer and V. Torra, "Ordinal, continuous and heterogeneous k-Anonymity through microaggregation," *Data Mining Knowl. Discovery*, vol. 11, no. 2, pp. 195–212, Sep. 2005.

[18] S. Martínez, D. Sánchez, and A. Valls, "A semantic framework to protect the privacy of electronic health records with non-numerical attributes," *J. Biomed. Informat.*, vol. 46, no. 2, pp. 294–303, Apr. 2013.

[19] S. Martínez, D. Sánchez, and A. Valls, "Semantic adaptive microaggregation of categorical microdata," *Comput. Secur.*, vol. 31, no. 5, pp. 653–672, Jul. 2012.

[20] R. Dingledine and N. Mathewson, "Tor: The second generation onion router," in *Proc. 13th USENIX Secur. Symp.*, Aug. 2004, pp. 1–5.

[21] S. J. Murdoch and P. Zielinski, "Sampled traffic analysis by Internet exchange-level adversaries," in *Proc. 7th Workshop Privacy Enhancing Technol.*, N. Borisov and P. Golle, Eds. Heidelberg, Germany: Springer-Verlag, Jun. 2007, pp. 1–5.

[22] A. Houmansadr and N. Borisov, "The need for flow fingerprints to link correlated network flows," in *Proc. 13th Privacy Enhancing Technol. Symp.*, vol. 7981. Berlin, Germany: Springer, 2013, pp. 205–224.

[23] S. J. Murdoch and G. Danezis, "Low-cost traffic analysis of tor," in *Proc. IEEE Symp. Secur. Privacy*, May 2006, pp. 183–195.

[24] X. Wang, S. Chen, and S. Jajodia, "Network flow watermarking attack on low-latency anonymous communication systems," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2007, pp. 116–130.

[25] X. Fu and Z. Ling, "One cell is enough to break tor's anonymity," in *Proc. Black Hat Tech. Secur. Conf.*, Feb. 2009, pp. 578–589.

[26] S. Chakravarty, M. V. Barbera, G. Portokalidis, M. Polychronakis, and A. D. Keromytis, "On the effectiveness of traffic analysis against anonymity networks using flow records," in *Passive Actours Measurement*, vol. 8362, M. Faloutsos and A. Kuzmanovic, Eds. Cham, Switzerland: Springer, 2014, pp. 1–5.

[27] M. Akhoondi, C. Yu, and H. V. Madhyastha, "LASTor: A low-latency AS-aware tor client," in *Proc. IEEE Symp. Secur. Privacy*, San Francisco, CA, USA, May 2012, pp. 476–490, doi: 10.1109/SP.2012.35.

[28] M. Edman and P. Syverson, "As-awareness in tor path selection," in *Proc. 16th ACM Conf. Comput. Commun. Secur.*, Chicago, IL, USA, 2009, pp. 380–389.

[29] K. Bauer, D. McCoy, D. Grunwald, T. Kohno, and D. Sicker, "Low-resource routing attacks against tor," in *Proc. ACM workshop Privacy Electron. Soc.*, 2007, pp. 11–20.

[30] N. P. Hoang and D. Pishva, "Anonymous communication and its importance in social networking," in *Proc. 16th Int. Conf. Adv. Commun. Technol.*, Pyeongchang, South Korea, Feb. 2014, pp. 34–39, doi: 10.1109/ICACT.2014.6778917.

[31] I. A. Gomaa, E. Abd-Elrahman, and M. Abid, "Virtual identity approaches evaluation for anonymous communication in cloud environments," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 2, pp. 367–376, 2016, doi: 10.14569/IJACSA.2016.070251.

[32] M. A. I. M. Aminuddin, Z. Zaaba, and A. Hussain, "Applicability of Website fingerprinting attack on Tor encrypted traffic," *Int. J. Innov. Technol. Exploring Eng.*, Vol. 8, no. 8, p. 3075, Jun. 2019.

[33] G. Montenegro and C. Castelluccia, "Statistically unique and cryptographically verifiable (SUCV) identifiers and addresses," in *Proc. 9th Annu. Netw. Distrib. Syst. Secur. Symp.*, San Diego, CA, USA, 2002, pp. 1–5.

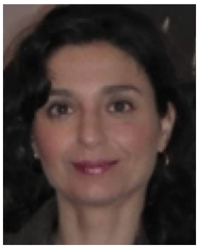
[34] J. Daniel. S. Bernstein, and T. Lange. *eBACS: ECRYPT Benchmarking of Cryptographic Systems*. Accessed: Aug. 18, 2020. [Online]. Available: https://bench.cr.yp.to,

[35] C. Fellbaum, *WordNet: An Electronic Lexical Database*. Cambridge, MA, USA: MIT Press. 1998.

[36] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "L-diversity: Privacy beyond k-anonymity," *ACM Trans. Knowl. Discovery Data*, vol. 1, no. 1, p. 3, 2007.

[37] N. Li, T. Li, and S. Venkatasubramanian, "T-closeness: Privacy beyond k-Anonymity and l-Diversity," in *Proc. IEEE 23rd Int. Conf. Data Eng.*, Istanbul, Turkey, Apr. 2007, pp. 106–115.

MERCEDES RODRIGUEZ-GARCIA received the B.Sc. degree in computer science from the University of Cádiz, Spain, the M.Sc. degree in ICT security from the Open University of Catalonia, Spain, and the Ph.D. degree in computer science and mathematics of security from Rovira i Virgili University, Spain. She is currently an Assistant Lecturer with the Department of Automation Engineering, Electronics and Computer Architecture, University of Cádiz. Her research and teaching interests include data privacy, computer network security, and reverse engineering and secure architectures.



MARÍA-ÁNGELES CIFREDO-CHACÓN received the B.Sc. degree in electronic engineering, the B.Sc. degree in industrial organization engineering, and the Ph.D. degree in industrial electronics from the University of Cádiz, Cádiz, Spain, in 1994, 1997, and 2010, respectively. She has been a Lecturer since 1998 and an Associate Professor since 2016 with the Department of Systems Engineering and Electronics, University of Cádiz. Her research interests include synthesis of electronic circuits from HDL descriptions and embedded processors in FPGA platform.

ÁNGEL QUIRÓS-OLOZÁBAL received the B.Sc. degree in electronic engineering from the University of Cádiz, Cádiz, Spain, in 1987, the B.Sc. degree in physics, specialized in electronics, from UNED, Madrid, Spain, in 1991, and the Ph.D. degree in industrial electronics from the University of Cádiz, in 2002. He has been an Assistant Professor of electronics from 1987 to 1996 and an Associate Professor of electronics since 1996 with the Department of Systems Engineering and Electronics, University of Cádiz. His research interests include boundary-scan test and synthesis of electronic circuits from VHDL description.