Scalable Role-based Data Disclosure Control for the Internet of Things

Smart Healthcare

Ali Yavari*†, Arezou Soltani Panah*, Dimitrios Georgakopoulos†,
Prem Prakash Jayaraman†, Ron van Schyndel*
*RMIT University, Melbourne, Australia
†Swinburne University of Technology, Melbourne, Australia
mail@aliyavari.com, arezou.soltanipanah@rmit.edu.au, dgeorgakopoulos@swin.edu.au,
pjayaraman@swin.edu.au, ron.vanschyndel@rmit.edu.au

***Abstract*—The Internet of Things (IoT) is the latest Internet evolution that interconnects billions of devices, such as cameras, sensors, RFIDs, smart phones, wearable devices, ODBII dongles, etc. Federations of such IoT devices (or *things*) provides the information needed to solve many important problems that have been too difficult to harness before. Despite these great benefits, privacy in IoT remains a great concern, in particular when the number of things increases. This presses the need for the development of highly scalable and computationally efficient mechanisms to prevent unauthorised access and disclosure of sensitive information generated by things. In this paper, we address this need by proposing a lightweight, yet highly scalable, data obfuscation technique. For this purpose, a digital watermarking technique is used to control perturbation of sensitive data that enables legitimate users to de-obfuscate perturbed data. To enhance the scalability of our solution, we also introduce a contextualisation service that achieve real-time aggregation and filtering of IoT data for large number of designated users. We, then, assess the effectiveness of the proposed technique by considering a health-care scenario that involves data streamed from various wearable and stationary sensors capturing health data, such as heart-rate and blood pressure. An analysis of the experimental results that illustrate the unconstrained scalability of our technique concludes the paper.**

I. Introduction

The IoT is fuelling a paradigm shift of a connected world in which everyday objects become interconnected and smart. While IoT supports a vast array of applications across a variety of domains [1], some of the data collected by IoT are sensitive and must be kept private. Examples of sensitive IoT data are physiological data collected by wearable or attached biomedical sensors or location data collected by GPS and mobile phones. Disclosure of such data creates opportunities for criminal activity, and can result in serious harm or even death. Thus, despite its benefits, IoT presents a significant challenge to security and privacy, which is exacerbated by the unprecedented scale of devices [2]. Traditionally, such security issues are addressed with the aid of encryption techniques. However, IoT devices are extremely limited in computational power and memory resources and therefore those techniques cannot be applied [3].

To address the mentioned issue, in this paper, we propose a novel obfuscation technique for IoT data that uses a combination of lightweight digital watermarking and scalable contextualisation. Digital watermarking is the practice of embedding extra information within digital content itself, which is also called host data, in a matter that does not interfere with the normal usage of host data [4]. Such techniques have been mainly used for digital right management of multimedia content. Our watermarking technique perturbs the sensitive data more or less depending on the disclosure privileges of the data requester. Therefore, better/more obfuscation can be provided for more sensitive data by increasing the intensity of the watermark. To the best of our knowledge, a little research, if any, has been conducted for perturbing sensitive IoT data using digital watermarking. In contrast to many other data obfuscation techniques, such as those described in [5] and [6], our obfuscation technique is reversible only by the authenticated users having the appropriate disclosure privilege(s). Since there is no information loss, in our approach, data can be freely modified and retrieved repeatedly having the right obfuscating parameters. In this regard, our technique is reminiscent of a role-based access control whereby only users who have matching roles can access the target data [5].

Another innovation included in this paper involves combining this watermarking technique with a highly scalable technique for contextualisation called ConTaaS [7]. ConTaaS-based contextualisation excludes irrelevant data from consideration and reduces data volume in large-scale IoT data management and analysis applications. This contextualisation-driven data reduction improves the scalability and performance of implementing security and privacy-preserving mechanisms in an IoT setting. Moreover, it reduces the amount of computation (often referred to as *reasoning*) required to understand and measure the corresponding privilege level for accessing each particular data point.

The main contributions of this paper include the following:

- Introducing a novel data obfuscation technique that combines contextualisation with digital watermarking based on the disclosure privilege of matching roles,

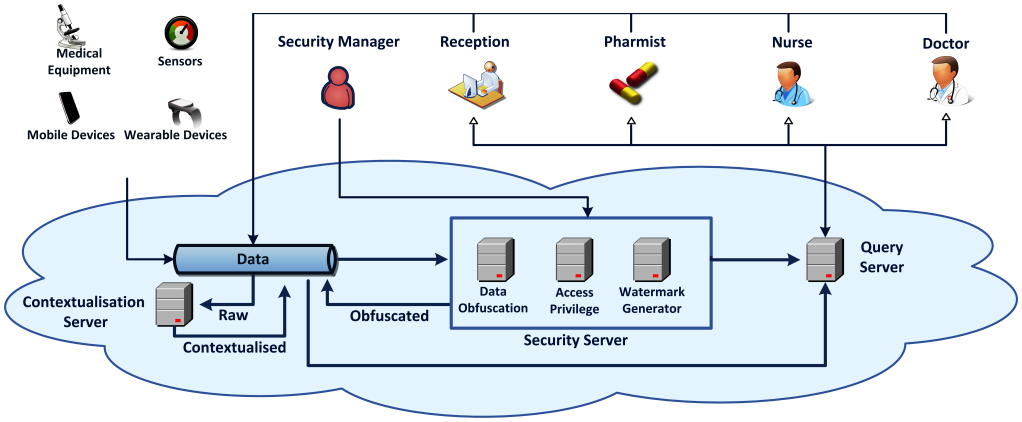

Fig. 1. Conceptual Architecture.

- Proposing a new Security-As-a-Service model that utilises this technique to govern access to IoT data (this will be discussed further in latter sections),
- Illustrating the scalability of the proposed technique by conducting an experimental evaluation for a health-care scenario to manage privacy-preserving access having 1000 IoT users/applications.

The remainder of the paper organised as follows. In Section II, we discuss related work. In Section III, we introduce our Security-as-a-Service model followed by the proposed watermark construction and obfuscation/de-obfuscation techniques. Section IV includes the description of a health-care use case we use in our evaluation. Section V presents our evaluation results. Finally, Section VI concludes the paper with future research directions.

II. Related Work and Background

There are several aspects of IoT that present security and privacy problems including IoT device communications, constrained resources (e.g. limited battery life), variety (e.g. different types of devices made by multiple manufactures), and the scale, i.e., billions of devices [8]. Among the plethora of recent research solutions [9], [10], [11], [12], [13] for protecting sensitive IoT data, some related research, e.g., in [14], [9], [11], focuses on security and privacy preservation policies while other related research, e.g., in [9], [10], [12], [13], focuses on encryption and the design of privacy preserved frameworks for IoT [15]. Although most of the these techniques can ensure security and privacy, their ability to scale-up for IoT devices and data has not been validated.

The scalability of privacy-preservation solutions is a grant IoT challenge. The solution proposed in this paper couples watermarking with contextualization to protect the privacy of a virtually unlimited number of IoT data points. In the following Sections we discuss further related work from the perspective of privacy preservation, digital watermarking, and contextualisation.

A. Related work

There are clear parallels between our proposed disclosure technique and access control. The main purpose of access control is to limit the actions or operations that a legitimate user of a system can perform, whereas disclosure control aims at publishing/sharing data such that the privacy of individuals is not compromised. There has been a a considerable volume of research on developing both access and disclosure control methods. The summary of the most related ones to our technique is given bellow.

The most common access control mechanisms are Discretionary Access Control (DAC), Lattice-Based Access Control (LBAC), and Role-based Access Control (RBAC) [16]. DAC is discretionary in the sense that the owner of the requested resource controls the access to that resource. Each access request is checked against the specified authorizations. If there exists an authorization stating that the user can access the resource in the specific mode (read or write), the access is granted, otherwise it is denied. LBAC enforces one-directional information flow on the basis of a predefined lattice of security labels which are associated with every resource and user in the system. RBAC determines access level via the role abstraction, rather than by just identity or clearance of the requester. In this model, a role is a semantic construct which is often a representation of a job in an organization.

In an IoT setting where both data and access control policies dynamically change, the above access models are not suitable to adopt these changes. In order to address such compliance requirements, the next line of research enrich access polices with contextual information. For instance, several extensions to the basic RBAC model are purposed to incorporate context variables such as Generalized RBAC (GRBAC) model [17]. GRBAC introduces environmental information such as temperature or location to *activate* roles based on the value of conditions in the environment where the request has been made. Likewise, a context-aware RBAC model was proposed for health-care applications [18], whereby the contextual information invokes the relevant access policies for a specific role. A major deficiency of these approach is that, data access is either granted or denied. In contrast, our role-based model is capable of granting *multi-granularity* access based on the privileges associated with the roles.

In order to provide flexibility for situations where different

granularity is needed, disclosure control methods are advantageous. We divide the existing disclosure control techniques into two main classes of identity and data disclosure control. The former techniques such as k-anonymity and l-diversity or pseudonymity attempt to detach or replace identifiers from data, whereas the latter techniques protect the data itself. We only discuss the data disclose control techniques in this work. A comprehensive review of the identity disclosure control techniques can be found here [19].

The common techniques for data disclosure control include, but not limited to, generalisation and suppression, data swapping, and noise addition. Data generalisation attempts to prevent data linkage for privacy preservation of published datasets. An example would be replacing the exact date of birth by only the year. Suppression techniques can be viewed as the ultimate generalisation since no information is released. Unfortunately, these techniques cause information loss, and also are not appropriate for real-time applications because of the complexity of the required calculations.

Our technique for data disclosure control is similar to noise addition techniques. For this purpose, we use digital watermarking techniques to obfuscate sensitive data. In contrast to the noise addition techniques, our technique is reversible which enables us to tune the obfuscation parameters based on the access privilege of the users.

B. Digital Watermarking

Digital watermarking is a proven technique in the multimedia domain for copyright protection [4]. The watermark constitutes a piece of secret information to be hidden within the digital content in such a way that it is not visible to the consumer. This requirement is called *invisibility*. Recently, there has been an explosion in non-media applications of digital watermarking among which are time-series, biological sequences, graphs, spatial, spatio-temporal, and streaming data [20]. In such applications, the watermark invisibility is no longer defined by human perception characteristics and often depends on specific application requirements. Since the focus of this work is IoT that typically generates data streams, we narrow down our attention to digital watermarking of streaming data such as sensory data.

Spread Spectrum (SS) is a popular approach for digital watermarking of sensory data, where a watermark is constructed as a random sequence that is *imperceptibly* inserted in a spread-spectrum-like fashion into the data values. Such sequences are often near-orthogonal codes of +1 and -1 symbols, and can be decoded through correlation between code pairs. The security of the SS watermarking technique is highly dependent to the spreading sequences. It is ideal to use truly random sequences so that no one other than the encoder could generate and predict the watermark. Unfortunately, the necessary hardware for generating such codes is not generally available [21]. Besides, if the decoder has to generate the same code to retrieve the encoded information, being truly random, the same code cannot be obtained. Instead, Pseudonoise sequences (PN) are used to resemble the random behaviour.

TABLE I
COMPARISON OF PN FAMILY SETS

Type	Length (l)	Maximum correlation bound	Family size	Normalized linear complexity
Gold	$2^n - 1$	$2^{(n+1)/2} - 1$ or $2^{(n+2)/2} - 1$	$l + 2$	$\frac{2n}{2^n - 1} \cong 0$
Small-Kasami	$2^{2n} - 1$	$\sqrt{l}$	$\sqrt{l}$	$\frac{1.5n}{2^{2n} - 1} \cong 0$
Large Kasami	$2^{4n+2} - 1$	$2\sqrt{l}$	$l \times \sqrt{l}$	$\frac{2n}{2^{2n} - 1} \cong 0$

Randomness is an ensemble property and cannot be achieved in a single sequence [22]. If an ensemble of PN codes are attempted to be encoded on the same data stream (either one data stream or an aggregated data stream such as moving average [23]), two other properties are need: high auto-correlation of a PN code and the low cross-correlation between any two PN codes in the same code family or set. Auto-correlation refers to the degree of correspondence between a code and a phase-shifted replica of itself. The cross-correlation is defined between two codes and represents the degree of agreements and disagreements between them.

An ensemble of periodic PN sequences with low off-peak auto-correlation and cross-correlation can be generated using maximal length sequences or m-sequences [22]. For example, in [24], an ensemble of l PN codes are shifted versions of a primitive m-sequence. Nearly n bits can be encoded through the phase, i.e., the number of spatial shifts (with a cyclic wrap-around), of a $l = 2^n - 1$. For increasing the number of possible PN codes, more primitive PN codes with low cross-correlation can be used. Two of known ensembles of such are Gold and Kasami [23]. Gold is a set of $2^n + 1$ sequences of length $l = 2^n - 1$, $(n \neq 4)$ whose cross-correlation are three valued. For n odd, the values are optimal and bounded by $2^{(n+1)/2} - 1$. Kasami codes of length $l = 2^n - 1$ only exist for even values of n. There are two classes of Kasami sequences namely Small set and Large set. The Small set has better correlation properties compared to the Gold and Large set. The summary of the described PN codes is listed in Table I. Linear complexity in this table refers to the security of PN codes against unauthorised detection.

To the best of our knowledge, there is no related work that makes use of digital watermarking for obfuscation of IoT data. The only similar work in a non-IoT setting has been proposed recently by Vlachos et al. [25] for right protection and data obfuscation of trajectory datasets. The proposed technique guarantees the preservation of hierarchical clustering operations after watermark insertion that is necessary for distance-based mining applications.

C. Contextualisation of IoT Data

In the IoT literature, context is defined as "any information about the entities (person, place or things) that are relevant to a given application(s) that can be used to contextualise data for that given service(s)" [7]. Contextualisation of IoT data for

supporting an IoT application is defined in [7] and involves the following three steps:

- Context Collection - User context information can be collected from user's smart-phone, wearable devices, or manually provided by the user. Moreover, cloud services can help to deduce new context information from the collected context.
- Contextualisation of the IoT data - We contextualise IoT data based on two main operations including Contextual Filter and Contextual Aggregation described in detail in [26]. Contextual Filter, filters the data originating from IoT devices and services based on the current context. Contextual Aggregation, aggregates the Contextual Filtered data based on the contextual similarities and relevance.

III. IoT Security Service

In this paper, we define (security) context as any information that can describe or impact the disclosure privilege of the data for the relevant roles, and provide IoT data security as a service. Our IoT Security Service includes aspects of confidentiality, controlled disclosure, authentication, and authorization. Its implementation involves using cloud computing infrastructure and service-oriented computing principles to provide this service for use by others. In the scope of this paper, we focus only on a contextualised authentication and data disclosure control for IoT data. Despite, the architecture is capable of performing other IoT security services as a service. The summary of notations that we use in the rest of the paper is given in Table II.

TABLE II
Notations

Symbols	Meanings
d	number of disclosure privilege regions
r_k	$k-th$ region index ($1 \leq k \leq d$)
rid_k	$k-th$ region identifier
key_k	The binding key for region rid_k
$skey_i$	The session key for user i
Ξ	An ensemble of PN sequences with desired correlation properties
l	The lenghth of PN sequence/code
σ	The composite template key
$rindex_o$	The region index attached to the data object O
u_i	i-th user/data-requestor
$Hash$	Hash functions, example SHA-1 or MD5
$h_{k,1}$, $h_{k,2}$	first half and the second half of the calculated hash for the region r_k
α_k	Scale factor (watermark amolitude) corresponds to the region r_k
$puKey_{DS}$	The public key of the Data Delivery service
$prKey_{DS}$	The private key of the Data Delivery service

A. Role-based disclosure privilege model

To explain our IoT Security Service, consider a nested role-based model of security privileges, where the least privilege is granted to the individuals located at the most inner region and the highest belongs to the individuals in the most outer region. This means an individual at a particular region, have all privileges of the regions they enclose as well. We denote the number of privilege regions (PR) by d, where d is the number of predefined roles in the system. Moreover, each PR has a unique identifier, denoted as rid, assigned to it. Therefore, a user with rid_k has all the disclosure privileges granted to all other users within regions rid_1 to rid_{k-1}. Please note that, the region identifiers are only known to the Security Service.

To grant access to data, our IoT Security Service exploits the knowledge of existing roles for authenticating users (who interact with the system by issuing queries). Therefore, every user belongs to a specific PR and the Security Service verifies this membership via a key that is assigned to every user. If the user's key is valid, the associated region id, i.e., rid, is retrieved to de-obfuscate the query result later.

To further explain our role-based model, we use the following notation: Every region r_k is associated with a pair (key_k, rid_k), where key_k is the secret key for all users belong to that region and rid is as defined above. Secrete keys and the corresponding region identifiers are generated by taking advantage a known ensemble construction, such as Kasami. More specifically, the binding key to a region is a PN code and the associated rid is the spatial shift value that can be used to generate other orthogonal PN codes as described in Section II-B. Next, we use these notation to describe in detail how these values are generated and how the Security Service can retrieve the associated rids without having access to the individual keys (i.e. the PN codes).

B. Conceptual Architecture

In the IoT, data typically is captured from various Internet-connected devices such as smart phones, wearable devices, and sensors. This data is often not protected. Applying traditional security techniques such as encryption is not feasible due to resource limitation of the IoT devices. In this paper, we aim to protect data from such IoT sources with a light-weight and scalable technique by using contextualisation and watermarking. Data disclosure control is achieved by a role-based privilege model described earlier.

Fig. 1, illustrates the conceptual architecture of our novel role-based data disclosure control. The primary components of this architecture are contextualisation, security and data delivery services. We used ConTaaS [7] to Contextual Filter and Contextual Aggregate the triples based on the their relevancy to the available roles. Contextualisation service deduces the associated access privilege (i.e. the label $rinedx_O$ that we describe in subsection III-E) for each individual data based on the privilege ontology[1] and the defined security requirements (e.g. policies defined by a security manager). Security service consisting of disclosure privilege, data obfuscation and watermarking is responsible for providing defined

[1]Ontology is a formal way of describing taxonomies and defining the structure of knowledge. Although, description of the privilege ontology is out of the scope of this paper, we refer to it as a knowledge repository describing the relation between the privacy-sensitivity of the data and the roles.

policies to contextualisation service, perturbing data and role-based authentication using watermark, respectively. Finally, data delivery is in charge of privacy-preserving delivery of the query result to the users. This conceptual architecture will be explained in a scenario in IV-A.

C. Data Model

In this paper we employ semantic web standards such as RDF and SPARQL to model and interchange data. The RDF format is N-triples [27]. A triple is a statement that describes data in the form of $< Subject, Predicate, Object >$. Subject is the identifier of the entity that the data is describing; Object is the description of the Subject in terms of the relation described in Predicate. For example a triple such as $< Patient1, hasHeartrate, 85 >$, describes that Patient1 heart-rate was 80. Furthermore, SPARQL is a query language for RDF triples.

D. Watermark Generation

Following the Security-As-a-Service notion, the watermark generation and exchange are delivered "as a service" to users in order to satisfy disclosure privilege requirements. Therefore, we make the assumption that, there is a trusted third party that knows *only* the summation of all shifted keys associated with all defined roles and thus can retrieve the rid of the data requester. In contrast, the contextualisation service is not trusted and therefore, only the obfuscated versions of data are stored in database (Subsection III-E).

Suppose we have an ensemble of PN sequences of length l (with low off-peak auto-correlation and cross-correlation), denoted as $\Xi = PN_1, PN_2, ..., PN_{|\Xi|}$. Examples of such ensembles are Gold and Kasami set. From this set, we choose a unique sequence $PN_j (1 \leq j \leq |\Xi|)$ as the key for all users in region r_k . On the server side, the received key is used to retrieve the associated rid_k. Recall this number is an integer value to shift the chosen PN_j and should be less the PN length i.e. $rid_k < l$; otherwise, the shifted PN codes will not be unique (because of the cyclic wrap-around). This value should be retrieved before granting data access to the user.

Apart from the PN codes which are identical for users in the same region, every user obtains a session key that makes the de-obfuscation process dependent on the his/her unique credentials and therefore enhances the security of our technique. This session key is generated by Security Service and exchanged using a secure exchange protocol such as SSL/TLS. We denote the session key for user i by $skey_i$.

The process for retrieving region ids (rids) is equivalent to de-spreading of the secret PN code (keys). This is done by a correlation operation between the template PN sequence and the received PN code from the user. The underlying principle behind decoding process is based on the observation that if in a cross-correlation between an embedded PN sequence and a template, the two differ only by a shift, then the correlation peak will be shifted by that amount. More detailed information about decoding process can be found here [23].

The template sequence, σ is a composite PN sequence obtained from the summation of several shifted versions of the original PN codes that are assigned to different regions, i.e. $\sigma = \sum_{i=1}^{d} shift(PN_i, rid_i)$, where $shift()$ represent a spatial shift with cyclic wrap around. Then, the periodic correlation is performed as $\rho(\tau) = \sum_{j=1}^{l-1} \sigma(j) PN_i(j + \tau)$. If PN_i is the correct key, the correlation values (ρ) reveal a significant peak at the position corresponds to rid_i. This value is passed to the data delivery service to de-obfuscate the data prior to sending it back to the user. It is clear that, if the key is not valid, then the retrieved rid will be incorrect which means the original information cannot be retrieved successfully.

Our proposed disclosure control has three main advantages:

- First, the PN codes can be generated on the fly in the most compact Linear Feedback Shift Register's using FPGA which is a lightweight and cost-effective approach.
- Second, storing one composite key instead of individual keys eases the key management burden at the server end and makes our scheme more scalable compared to storing different keys for different users. This additionally increases the security of our scheme, if Security Service is compromised.
- Third, the use of session keys affords us the ability to have a fine-grained disclosure privilege for authenticated users.

E. Data Obfuscation

Before explaining the obfuscation process, it is important to remember, after contextualisation a hierarchy of data is built upon desirable privilege. For this purpose a set of privilege policies are required such as 'The ECG data can only be accessed by Doctors', or 'The blood pressure data can be accessed by Nurses and Doctors'. Based on these policies and the role-based privilege model, the contextualisation service attaches a tag to the data. We assume, for every data object O, the region index $rindex_O$ is attached to that datum.

Based on this assumption, the data storage might be outside of the trust surface, therefore we modify the original values using an obfuscation function (OF) in a way that only the authenticated users with the right privilege could de-obfuscate data and retrieve the original values. In the literature, there are many OF for this purpose. As already discussed in Section II, random noise addition generated from a probabilistic distribution such as Laplacian is a possible candidate for OF. However, the use of truly random numbers makes the de-obfuscation process non-reversible. Since here we are concerned with highly sensitive biomedical data, a reversible OF is desired. Therefore, here, a *deterministic* OF by means of digital watermarking techniques is used to provide a reversible obfuscation transformation.

We take advantage of an additive watermarking approach, whereby the obfuscated data is simply constructed by adding a scaled watermark to the data. Following our notations, the watermarked data is obtained as $O^w = O + scale(w)$. Traditionally, $scale()$ updates the amplitude of the watermark w to make it imperceptible from the host data. In contrast,

TABLE III
OBFUSCATION PARAMETER TABLE

region index	roles	region id	scaling	first half of hash	second half of hash
1	role 1	rid_1	α_1	$h_{1,1}$	$h_{1,2}$
2	role 2	rid_2	α_1	$h_{1,1}$	$h_{1,2}$
.	.	.	.	.	.
k-1	role k-1	rid_{k-1}	α_{k-1}	$h_{k-1,1}$	$h_{k-1,2}$
k	role k	rid_k	α_k	$h_{k,1}$	$h_{k,2}$
k+1	role k+1	rid_{k+1}	α_{k+1}	$h_{k+1,1}$	$h_{k+1,2}$
.	.	.	.	.	.
.	.	.	.	.	.
d	role d	rid_d	α_d	$h_{d,1}$	$h_{d,2}$

here, we can embed watermarks with larger scale values if better obfuscation is desired.

In our design, we add two more amendments to the above watermark encoding to make the obfuscation technique dependable on the users' privileges. First, the embedded watermark is a keyed hashed value of the retrieved rid with the composite key σ. This makes the data obfuscation dependable on the Security Service and prevents calculation of the hash, if an adversary intercepts the rid. Second, the watermark amplitude is tuned in such a way that for data with higher sensitivity, the scale factor is larger. The rationale behind this is that, such data presents more sensitive information about the individuals and therefore the privacy is much more important compared to the data that can be accessed with the lower disclosure privilege. Since, our de-obfuscation technique is reversible, we can add a large amplitude of watermark to the original data and later on subtract the added value to retrieve the original data. Again these values should be selected a prior based on the desired privilege policies.

In summary, the obfuscated or watermarked data object is generated as $O^w = O + \alpha_k \times decimal(Hash(rid_k, \sigma))$, where α_k is the associated scale factor for the region r_k and $decimal()$ returns the decimal hash value. Once a query is issued, the Security Service retrieves the parameters for that user and passes it to the data delivery service to de-obfuscate the data. However, this approach still has one problem. For those data that can be accessed from multiple regions, the data delivery server cannot distinguish the obfuscating parameters and consequently, invalid values will be reported.

For solving this problem, we change the watermark value and store an extra table (such as Table III) in the Security Service. First the above hash value, $Hash(rid_k, \sigma)$ is split into half, say $h_{k,1}$ and $h_{k,2}$. The data is, then, obfuscated by the scaled version of the first half i.e. $O^w = O + \alpha_k \times decimal(h_{k,1})$. The second half, $h_{k,2}$, replaces the original region index $index_O$, that is later used to find out the disclosure privilege of that data. Therefore, the obfuscated data can be represented by the quadruplet $< S, P, O^w, h_{k,2} >$. This means for data de-obfuscation, the associated values including rid, α, h_1, and h_2 are required that needs to be stored.

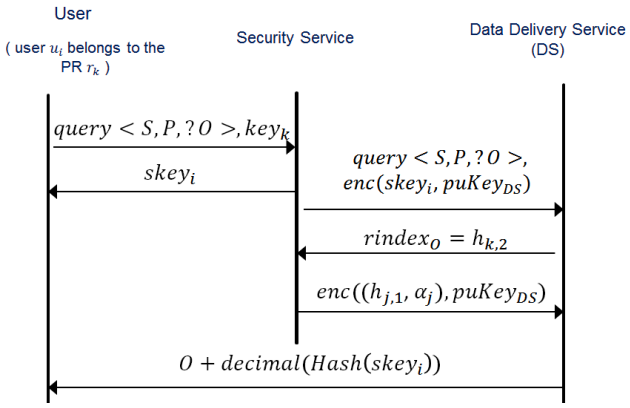

Fig. 2. Sequence diagram of data delivery

F. Data Delivery

At this step, the query server de-obfuscates the data based on the received parameters from the Security Service. Then, the data delivery service re-obfuscates data using the session key before sending the result to the user. From the technical standpoint, this not only limits data disclosure at rest, but also while being transmitted to the data requester,

Assume, a user u_i belongs to the region r_k. The entire process is described step-by-step as follows:

1) The user u_i sends its query for data object O i.e. ($< S, P, ?O >$), along with its secret (PN code) to the Security Service,
2) If the key is correct, the Security Service retrieves the associated rid_k and creates a session key, say $skey_i$, and sends back a copy of the session key to the user. Also another copy of the session key is created using the public key of the data delivery service (i.e. $enc(skey_i, puKey_{DS})$, where $enc()$ is an encryption function) and is sent along with the query to the Data Delivery Service,
3) The Data Delivery Service sends the corresponding $rindex_O$ for the requested Object O to the Security Service, which is effectively $h_{k,2}$.
4) The Security Service looks up into the Obfuscation Parameter table for the equivalent hash value and retrieves the corresponding $h_{j,1}$, α_j values. Then these values are again encrypted with the public key of the data delivery service and sent to the data delivery service,
5) The Data Delivery Service, consequently decrypts the received information using its private key $prKey_{DS}$ to extract the scaling factor and the watermark and subtracts the multiplication of the two values from the obfuscated data,
6) The data delivery service re-obfuscates the data using the session key, before sending it to the user. For this purpose, he hash value of the session key is calculated and its decimal value is added to the original data,
7) Finally, the user u_i de-obfuscates the data by subtracting the hash value of the session key and obtains the original data.

The aforementioned steps are illustrated in the Fig. 2.

IV. Implementation Scenario

Smart-health is a new paradigm of using the IoT and recent technologies such as wireless sensor networks and cloud computing towards delivering smart health care services to the citizens [28]. These services require accessing to the private and privacy-sensitive data from the citizens. However, protecting the privacy is challenging due to the limitations in processing capabilities of IoT devices on the one hand, and the enormous amount of data should be considered on the other hand.

The experimental scenario of this paper is an extension of the scenario described in [7]. Consider the outbreak of an epidemic disease such as Ebola. In order to control the disease, it is necessary to check and monitor the symptoms for all the citizens continuously and as quickly as possible. In our experimental scenario, not only the medical staffs are contributing in measurement of the symptoms but also smart devices (e.g. smart watch) can send data to the IoT applications. We have defined four roles with different access privileges as shown in Fig. 3.

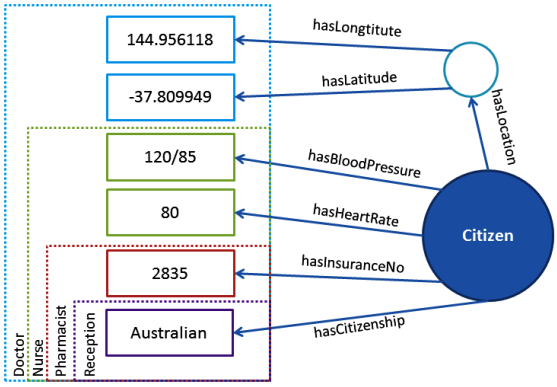

Fig. 3. Role-based privilege model for the health-care scenario.

A. Scenario

Contextual Filter will exclude irrelevant data for the queries. For example, if there is no interest to people who never had a heart-rate more or less than a particular value or people without any hear-rate record, those people will not be considered for any queries. Next, Contextual Aggregation will generate aggregated nodes as described in [7]. For example, if all the queries that were interested for heart-rate more than a particular value also were interested for blood pressure less than a particular value.

B. Test-bed

The experimental test-bed has been developed on Amazon EC2, "M4 General Purpose" instance, with 32 GB RAM and 8 vCPU. Our synthesized RDF Dataset consists of data such as blood pressure, heart-rate, and location of 500 users captured every 10 minutes for 15 days while data regarding the insurance information and citizenship assumed to be entered into the database via medical or administration staff.

V. Evaluation

A. Performance

Fig. 4 shows the results of a performance evaluation we conducted over 16 days of data collection from patients. The collected data are stored in form of triplets represented on the horizontal axis of the graph. For example, 7200 samples were collected on the first day, performing contextual filtering took 66 ms, while contextual aggregation took only 11 ms. Similarly, watermark insertion took 1996 ms, watermarking combined with contextual filtering 80 ms, and watermarking combined with contextual aggregation only 18 ms. The experimental data clearly reveal the lightweightness of our watermarking technique, the effectiveness of contextualisation technique and the superiority of their combination (only 284 ms for processing 1152000 data points).

B. Remaining issues

From the security standpoint, our method has two potential problems. First, the usage of an ensemble of PN sequences as authentication keys resolves the problem of generating keys for computationally-bounded IoT devices, but it opens up the possibility of a Brute-force attack for guessing the secret rid values. Here, we used Small Kasami that gives us $2^{2n} - 1$ possible values for region IDs that is not many. However, this problem can be solved by using a more secure PN codes with larger set size, such as Moreno-Tirkel sequences [29], without changing the proposed technique.

The second problem is related to the watermark amplitudes for data obfuscation process i.e. α values. Here, we used a constant value to amplify the embedded watermark that makes our scheme vulnerable against the Wiener attack in which an attacker can remove the watermark by using statistical estimation. In order to combat this attack, the power spectrum of watermark should resembles that of the data, named here [30] as power-spectrum condition. This feature can be easily added to our existing model during the contextualisation process and makes our watermarks robust against this attack. We are already working on this improvement, but this is outside the scope of this paper.

VI. Conclusion

In this work, we introduced a role-based disclosure control technique suitable for any IoT application in which the dissemination of IoT data may violate the privacy of the individuals whose IoT devices contributed such data. Even if privacy preservation approaches have already been proposed for some specific IoT applications, a comprehensive architecture, general and flexible enough to deal with IoT constrained environments in a real setting, is still missing. For this purpose, we combined digital watermarking with contextualisation into an unified architecture that fulfills the function of a Security-as-A-Service. This service contextualises sensitive data to reduce data size prior to data obfuscation. Reversibility of the obfuscated data is also provided to users with the appropriate disclosure privileges. Therefore, only the perturbed versions of the original data is available to the public. When a query

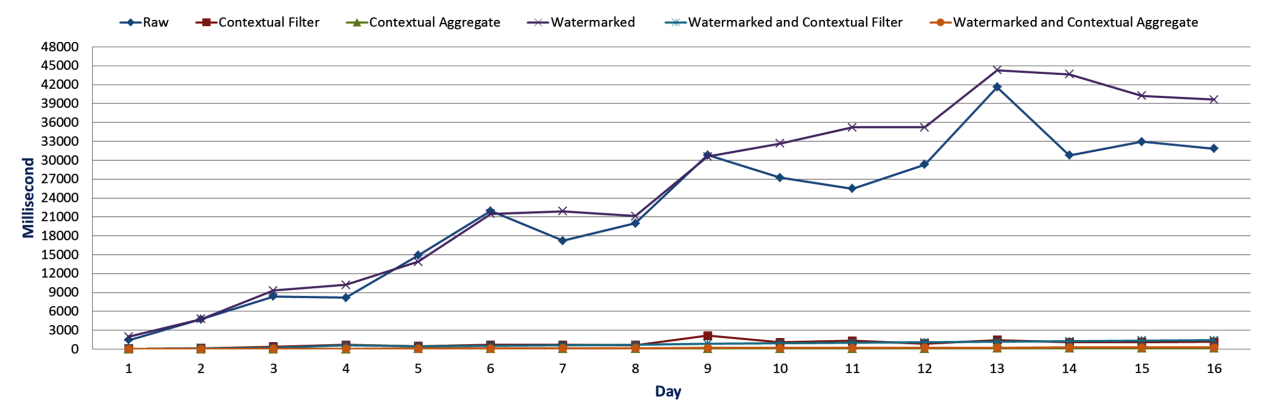

Fig. 4. Query Response Time

is issued, the de-obfuscation parameters are retrieved based on the provided credentials to retrieve original data values. In order to protect the original data during a transmission, such data is re-obfuscated so that only an authorized user can retrieve the original data. This way, the sensitive data can be protected while being transmitted and at rest. The results confirmed that the computational complexity of our proposed disclosure control is very modest.

REFERENCES

[1] A. Whitmore, A. Agarwal, and L. Da Xu, "The internet of thingsa survey of topics and trends," *Information Systems Frontiers*, vol. 17, no. 2, pp. 261–274, 2015.

[2] P. P. Jayaraman, X. Yang, A. Yavari, D. Georgakopoulos, and X. Yi, "Privacy preserving internet of things: From privacy techniques to a blueprint architecture and efficient implementation," *Future Generation Computer Systems*, 2017.

[3] D. E. Bakken, R. Parameswaran, D. M. Blough, A. A. Franz, and T. J. Palmer, "Data obfuscation: Anonymity and desensitization of usable data sets," *IEEE Security and Privacy*, vol. 2, no. 6, pp. 34–41, 2004.

[4] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital watermarking and steganography*. Morgan Kaufmann, 2007.

[5] E. Bertino, B. C. Ooi, Y. Yang, and R. H. Deng, "Privacy and ownership preserving of outsourced medical data," in *21st Int. Conf. on Data Engineering*. IEEE, 2005, pp. 521–532.

[6] J. Li, Y. Tao, and X. Xiao, "Preservation of proximity privacy in publishing numerical sensitive data," in *Proc. of the 2008 ACM SIGMOD Int. Conf. on Management of data*. ACM, 2008, pp. 473–486.

[7] A. Yavari, P. Jayaraman, D. Georgakopoulos, and S. Nepal, "Contaas: An approach to internet-scale contextualization for developing efficient internet of things applications," in *Proc. of the 50th Annual Hawaii Int. Conf. on System Sciences*. IEEE, 2017.

[8] R. H. Weber, "Internet of things–new security and privacy challenges," *Computer Law & Security Review*, vol. 26, no. 1, pp. 23–30, 2010.

[9] J. Hahn, "Security and privacy for location services and the internet of things," *Library Technology Reports*, vol. 53, no. 1, pp. 23–28, 2017.

[10] A. Ouaddah, A. A. Elkalam, and A. A. Ouahman, "Towards a novel privacy-preserving access control model based on blockchain technology in iot," in *Europe and MENA Cooperation Advances in Information and Communication Technologies*. Springer, 2017, pp. 523–533.

[11] L. A. Martucci, S. Fischer-Hübner, M. Hartswood, and M. Jirotka, "Privacy and social values in smart cities," in *Designing, Developing, and Facilitating Smart Cities*. Springer, 2017, pp. 89–107.

[12] E. Tragos, A. Fragkiadakis, V. Angelakis, and H. C. Pöhls, "Designing secure iot architectures for smart city applications," in *Designing, Developing, and Facilitating Smart Cities*. Springer, 2017, pp. 63–87.

[13] A. Bera, A. Kundu, N. R. De Sarkar, and D. Mou, "Experimental analysis on big data in iot-based architecture," in *Int. Conf. on Data Engineering and Communication Technology*. Springer, 2017, pp. 1–9.

[14] A. Rayes and S. Samer, *Internet of Things from Hype to Reality: The Road to Digitization*. Springer, 2016.

[15] Y. Ould-Yahia, S. Banerjee, S. Bouzefrane, and H. Boucheneb, "Exploring formal strategy framework for the security in iot towards e-health context using computational intelligence," in *Internet of Things and Big Data Technologies for Next Generation Healthcare*. Springer, 2017, pp. 63–90.

[16] E. Bertino and R. Sandhu, "Database security-concepts, approaches, and challenges," *IEEE Trans. on Dependable and secure computing*, vol. 2, no. 1, pp. 2–19, 2005.

[17] M. J. Covington, W. Long, S. Srinivasan, A. K. Dev, A. Mustaque, and G. D. Abowd, "Securing context-aware applications using environment roles," in *Proc. of the 6-th ACM symposium on Access control models and technologies*. ACM, 2001, pp. 10–20.

[18] J. Hu and A. C. Weaver, "A dynamic, context-aware security infrastructure for distributed health-care applications," in *Proc. of the first workshop on pervasive privacy security, privacy, and trust*. Citeseer, 2004, pp. 1–8.

[19] C. C. Aggarwal and S. Y. Philip, "A general survey of privacy-preserving data mining models and algorithms," in *Privacy-preserving data mining*. Springer, 2008, pp. 11–52.

[20] A. Soltani Panah, R. Van Schyndel, T. Sellis, and E. Bertino, "On the properties of non-media digital watermarking: a review of state of the art techniques," *IEEE Access*, vol. 4, pp. 2670–2704, 2016.

[21] S. Miller and D. Childers, *Probability and random processes: With applications to signal processing and communications*. Academic Press, 2012.

[22] A. Leukhin and A. Tirkel, "Ensembles of sequences and arrays," in *2015 Int. Workshop on Signal Design and its Applications in Communications*. IEEE, 2015, pp. 5–9.

[23] A. Soltani Panah, R. van Schyndel, T. Sellis, and E. Bertino, "In the shadows we trust: a secure aggregation tolerant watermark for data streams," in *IEEE 16th Int. Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, 2015, pp. 1–9.

[24] M. Maes, T. Kalker, J. Haitsma, and G. Depovere, "Exploiting shift invariance to obtain a high payload in digital image watermarking," in *IEEE Int. Conf. on Multimedia Computing and Systems*, vol. 1. IEEE, 1999, pp. 7–12.

[25] M. Vlachos, J. Schneider, and V. G. Vassiliadis, "On data publishing with clustering preservation," *ACM Trans. on Knowledge Discovery from Data (TKDD)*, vol. 9, no. 3, p. 23, 2015.

[26] A. Yavari, P. P. Jayaraman, and D. Georgakopoulos, "Contextualised service delivery in the internet of things: Parking recommender for smart cities," in *IEEE 3rd World Forum on Internet of Things*. IEEE, 2016, pp. 454–459.

[27] D. Beckett and A. Barstow, "N-triples," *W3C RDF Core WG Internal Working Draft*, 2001.

[28] T. Shah, A. Yavari, S. S. Karan Mitra, P. P. Jayaraman, F. Rabhi, and R. Ranjan, "Remote health care cyber-physical system: quality of service (qos) challenges and opportunities."

[29] O. Moreno and A. Tirkel, "New optimal low correlation sequences for wireless communications," in *Int. Conf. on Sequences and Their Applications*. Springer, 2012, pp. 212–223.

[30] J. K. Su and B. Girod, "Power-spectrum condition for energy-efficient watermarking," *IEEE Trans. on Multimedia*, vol. 4, no. 4, pp. 551–560, 2002.