SecP2I : A Secure Multi-party Discovery of Personally Identifiable Information (PII) in Structured and Semi-structured Datasets

1st Amine MRABET
Umanis, Levallois-Perret, France
amrabet@umanis.com

2nd Mehdi BENTOUNSI
Umanis, Levallois-Perret, France
mebentounsi@umanis.com

3rd Patrice DARMON
Umanis, Levallois-Perret, France
pdarmon@umanis.com

***Abstract*—Personal data governance is became a key issue within organisations. This is mainly due to *(i)* the strategic value of personal data which provide more insights improving commercial and operational efficiency ; and data security risk issues and privacy regulation restrictions (GDPR, CCPA, . . .). Creating data catalogs is an important step for setting up a personal data governance. However, it remains a time-consuming task especially because of the absence of naming conventions in database modeling coupled to the heterogeneity of database management systems (DBMS) across Information Systems (IS).**

The paper presents SecP2I, an efficient data analytics-based approach permitting personal data discovery in structured and semi-structured datasets while guaranteeing end-to-end data confidentiality. The effectiveness of the platform is proven using a real world HR dataset.

***Index Terms*—Personally Identifiable Information, Secure Multi-party Computation, Data Governance**

I. Introduction

Personally identifiable information (PII) also known as *personal data* means any information relating to an identified or identifiable person who can be identified directly by his name and first name or indirectly by his date of birth, home town, gender, . . . etc. Establishing a personal data governance ensures data owners to provide reliable, useful and accessible data in a secure way [9]. It also allows to comply with privacy regulation restrictions as GDPR and CCPA [8]. Thus we are seeing the emergence of innovative solutions in the market permitting to meet the challenges of personal data governance within the organizations [2], [10].

Automatic personal data discovery is the first step for building successful data governance strategies by Chief Data Officers and/or Data Protection Officers. The purpose is to provide centralized data catalogs within organizations. This need is mainly due to the increase of data variety and volume (i.e., big data) and the absence of documentations and data models for databases and applications composing the Information System.

There are many data discovery approaches in the literature, and we need to distinguish between two main categories:

1) Metadata scanning-based solutions are able to extract databases schemes in order to build metadata inventories (i.e., attribute names, data types and constraints) [3]. Lexical and semantic matching are then done on metadata in order to identify personal data in related datasets. Such approaches are often inefficient, especially with the absence of naming conventions. Consequently, they require an additional and manual time-consuming semantic enrichment by operators.
2) Data analytics-based solutions perform lexical and semantic analyzes on data stored in databases and applications [7]. Such data-based solutions are more efficient. However, confidentiality issues may occur because of sensitive data access and analytics.

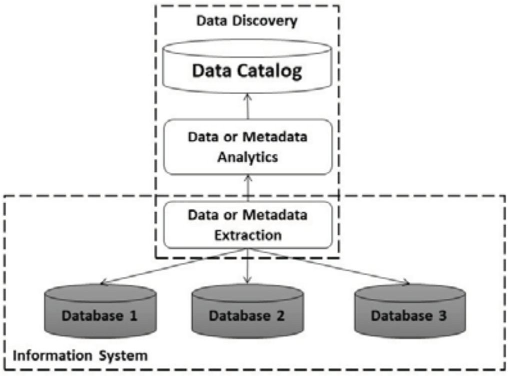

Fig. 1. Data Discovery Approaches and Security

As depicted in Figure 1, the security architecture is based on the security perimeter of both Information System and Data Discovery Platform, and the security of communication channel. This conventional security architecture ensures the security against only external attacks. However, in a multi-party architecture, computations could occur between partially trusted partners [4]. Indeed, sensitive data about identity, health, . . . are extracted from databases and disclosed to the

978-1-7281-0858-2/19/$31.00 ©2019 IEEE

data discovery platform and also external operators. This problem is referred to as Secure Multi-Party Computation (SMC) problem in the literature [11]. SMC enables distributed parties to jointly compute an arbitrary functionality without revealing their own private inputs and outputs.

To address these issues, we provide SecP2I, a privacy-preserving PII discovery platform in structured and semi-structured datasets. SecP2I is based on a SMC protocol to ensure the confidentiality of personal data in Information System against curious adversaries. Our contributions can be summarized as follows :

1) We define a security model of multi-party architecture for PII discovery taking into account semi-honest third party attacks.
2) We provide a secure multi-party computation protocol to ensure the confidentiality of data in Information Systems against curious adversaries. The protocol is based on Private Set Intersection (PSI) problem to securely compute a matching rates between private datasets and external open data (names, first names, cities, pathologies,) [5].
3) We also use a privacy-friendly queries on datasets in order to calculate matching rates based on regular expressions and aggregation functions (Email, phone numbers, login, ...).
4) We evaluate the efficiency and the performance of the approach using structured (relational database: MySQL) and semi-structured (NoSQL document- oriented database: MongoDB) HR datasets provided by Teambuilder platform [6].

The paper is organized as follows. In Section 2, we formalize the problem and provide the security definition. A formal and technical descriptions of SecP2I are given respectively in Sections 3 and 4. Section 5 evaluates the performance of the platform through a real world HR data and concludes the paper.

II. Formalization and Security Definition

SecP2I can be defined as a dataset coupled with a set of reference vectors and regular expressions. Formally,

Definition 1. (Data vector) A data vector is a named and typed sequence of data values.
A data vector is a tuple $d = \langle V_i, N, T\rangle$ where: $V_i = (v_0, v_1, ..., v_{i-1})$ are data values. N is a character string indicating the name of the vector also called attribute. T is a character string coupled with a length indicating the type of data values.

Definition 2. (Dataset) Given a dataset D with a set of data vectors $\{d_1, d_2, \ldots, d_n\}$, $p\,[d_i]$ refers to the value of attribute d_i for the person p. Attributes of a dataset D are divided as follows:

- Personally identifiable information PII relating to an identified or identifiable person who can be identified directly by his name and first name or indirectly by his date of birth, hometown, gender, . . . etc
- Non personally identifiable information $NPII$ relating to information not falling under the above category.

Definition 3. (Reference vector) A reference vector represents a sequence of transformed PII using a transformation function. A reference vector is a tuple $d = \langle V_i, N, T\rangle$ where: $V_i = (v_0, v_1, ..., v_{i-1})$ are data values. N is a character string indicating the name of the vector also called attribute. T is a character string coupled with a length indicating the type of data values.

Definition 4. (Adversary Model) We consider in this paper curious adversary model. Curious adversaries can eavesdrop on various components of the system. These include: (i) the monitoring events table which contains all information that the EMS stores about the consumer, (ii) the communication channel which has all information sent between the consumer and curator, and (iii) the result of the processing. We consider adversaries that can eavesdrop on consumer components as outside of our attack model.

Definition 5. (Keyed-hash Functions) [1]
A hash function is a computationally efficient function that maps bitstrings of arbitrary length to bitstrings of fixed length, called hash values. Let K denote a $n-$dimensional vector space over $GF(2)$. A keyed-hash function $hash_k : k \in K; hash_k(m) = \acute{m}$ is indexed by a key k.

In the following, we describe some properties :

- Keyed-hash functions, e.g., MD5 or SHA-1, are primarily designed to be *collision resistant* ; hence, $hash_k(m_1) = hash_k(m_2)$, but $m_1 \neq m_2$.
- Given key $n \in K$ and message m , it is *straightforward* to compute $\acute{m} = hash_k(m)$.
- *Independence of input/output*: Given (possibly many) pairs of message m and result $hash_k(m)$, it is hard to find key k.

Definition 6. (Security Definition) To ensure *confidentiality* we will show that the protocol uses a cryptosystem without a key exchange, and the key-pair is stored inside the process consumer security perimeter. Moreover, complex monitoring event processing is done over encrypted component-identifiers and the adversary should recover the key-pair or cryptanalysis the system in order to infer them.

Namely, two parties, a client and a server, want to jointly compute the intersection of their private input sets in a manner that at the end the client learns the intersection and the server learns nothing .

III. Proposed approach

In this section we present our optimization objectives in the implementation of our secure model. We used several methods to launch the secure discovery on several levels.

A. Objectives

To implement our security model of multi-party architecture for PII discovery, we also sought to optimize on three challenges (Precision, Confidentiality and performance). The first one is to increase the detection accuracy. In order to achieve this objective, we propose several detection alternatives, a dynamic knowledge base, reference database and regular expressions.

A second objective is to guarantee the confidentiality of the data. Our solution offers secure detection. In our different detection techniques, we have no access to data in the clear.

The last goal is to provide a quick detection solution. To achieve this goal, we proposed a dynamic architecture of knowledge base. Our knowledge base is used to detect data at the metadata level. With this solution, we gain performance time by reducing data-level access, but we also gain privacy.

B. Detection levels

Our detection solution is at two levels: the level of metadata and the level of data. At the metadata level, we use an knowledge base. In this article, we will only discuss the training phase for the metadata level.

For the data level, we use two detection techniques. These techniques are secure detection by reference databases and secure detection by regular expression queries.

C. Detection Methods

Currently we propose three methods used for detection and for learning. These three methods are detection by knowledge base, detection by injection of regular expressions in our SQL and NoSQL queries, and detection based on a reference base.

D. Scenario

We propose two possible scenarios. A first scenario concerns the detection of personal data in a real customer database. A second scenario concerns the learning of our knowledge base (dynamic). For learning, we use real databases to provide an accurate score for detection. Algorithm 1 presents the first detection scenario. This algorithm has as inputs a client database, a static knowledge base, a dynamic knowledge base, access to the reference database and access to the list of regular expressions. The output of this algorithm is the list of attributes detected as personal data.

Our algorithm 1 starts by extracting the metadata from the database. Then it enchain the methods proposed in section III-C for detection.

We start the detection with the method based on dynamic knowledge base. Then using respectively the method based on regular expressions and the method based on reference bases. In order to better optimize the execution time we propose an optimization in the detection via the method of regular expressions. The latter based on a classification according to the types of data detected.

On the other hand, algorithm 2 presents a second scenario, it is the learning scenario. This learning algorithm takes the same inputs as algorithm 1. But the output of this algorithm is to update our dynamic knowledge base. The first input of this algorithm is $d[n]$. $d[n]$ is a list of the data vectors to be tested. The second is $d_{ref}[k]$, this is the list of reference vectors

This learning algorithm proposes three main functions are the search if an attribute exists or not, the insertion of a new attribute and the update of the attribute if it already exists in the knowledge base.

Algorithm 1: PERSONNEL DATA DETECTION

```
Input: $d[n] = \langle V_i, N, T \rangle$, $d_{ref}[k] = \langle V_j, N, T \rangle$,
       $knowledgeDB\_S$, $knowledgeDB\_D$,$Base\_RegEx$
Output: $PIIs$
$result$ ←Matching_knowledgeDB($d[n].N, knowledgeDB\_D$);
if ($d[n].N \bigcap result \neq Null$) then
    $PIIs$ ←Search_PII ($Result, knowledgeDB\_S$);
    return $PIIs$;
else
    foreach $x \in d[n].N \bigcap PIIs$ do
        $PII$ ←Detection_PII ($x, Base\_RegEx$);
        if ($PII \neq Null$) then
            $PIIs.add(PII)$;
        else
            $PII$ ←Detection_PII($x, d_{ref}[k].N$);
            if ($PII \neq Null$) then
                $PIIs.add(PII)$;
            else
                $Not\_Detected.add(PII)$;
return $PIIs, Not\_Detected$;
```

Algorithm 2: SCORING - IDENTIFICATION SCORE

```
Input: $d[n] = \langle V_i, N, T \rangle$, $d_{ref}[k] = \langle V_j, N, T \rangle$,
       $Base\_RegEx$, $knowledgeDB\_D$
Output: $UPDATED\_knowledgeDB$
foreach $x \in d[n].N$ do
    $PII$ ←Detection_PII ($x, Base\_RegEx$);
    if ($PII \neq Null$) then
        $PII\_O$ ←Search_DCP
            ($PII, knowledgeDB\_D$);
        if ($PII\_O = Null$) then
            Insert($PII, knowledgeDB\_D$);
        else
            Update($PII, knowledgeDB\_D$);
    else
        $PII$ ←Detection_PII($x, d_{ref}[k].N$);
        if ($PII \neq Null$) then
            $PII\_O$ ←Search_DCP
                ($PII, knowledgeDB\_D$);
            if ($PII\_O = Null$) then
                Insert($PII, knowledgeDB\_D$);
            else
                Update($PII, knowledgeDB\_D$);
        else
            $Not\_Detected.add(PII)$;
```

IV. Proposed Architecture

Figure 2 of this paper presents the architecture that translates the learning algorithm 2. In this architecture we present our knowledge base composed by two components, a static component contains the personal data defined by the law, for each attribute we define several information like the level of sensitivity, category and type. And a dynamic component, in this part we manage identification scores for the attributes detected during learning. Each attribute can be identified by several personal data in the static component. The identifications scores are calculated according to the number of the data detected during the learning and the score which already exists for each attribute (the score is initially null) We present in this architecture also a component of extraction of the metadata. This component performs several tasks such as cleaning, transforming data into uppercase and then storing the complete schema. And finally we have both detection components, using the regular expressions and the reference base. For regular expressions we inject these expressions into our queries. These provided the detection scores as a return result. For the reference base method, we initiate a process that cleans up, transforms to uppercase, and then cipher client data. With this process we protect the customer data and we guarantee that the data will never be processed in clear. To recover the detection score we propose matching with encrypted data.

V. Experimental Results

[6]

Our solution via this personal data detection approach is to provide scores to get an accurate estimate. The results we obtain allow decision and / or decision support. We obtain an automatic decision if a data is personal or not. Our tool offers recommendations for anonymizing data. In other words, depending on the parameters (category, sensitivity and type) of each type of data, we recommend certain anonymization proposals.

For example, we present the detection scores presented in Figure 3, which presents the scores detected by the method of the base of reference. These scores are called identification scores. In this example, we compare the first name, last name, street, and city with a references. In our detection, we used French data.

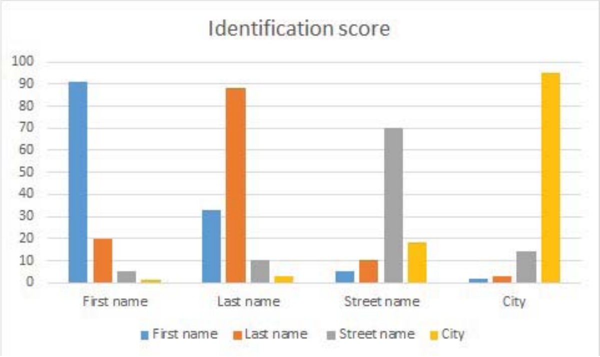

Fig. 3. Score by Method - Reference Bases

VI. Performance

As presented in the objective section III-A, we propose a solution with three objectives (Performance, precision and confidentiality). in this section we will discuss the performance objective. for this purpose we have launched performance analysis for the execution time. we test the performances for the methods RegEx and reference base. Our tests are done on structured database and semi-structured database. the graphs in Figures 4 5 et 6 present comparisons between true tests and false tests with SQL and NoSQL databases. The tests proposed in this paper are for the following attributes: First name, Last name, city, registration number and email. In Figure 4 we present the detection results in an SQL database. In this case we use MD5 for protection. In Figure 5 we present the detection results in an SQL database. In this case we use SHA1 for protection.

In Figure 6 we present the detection results in an NoSQL database. In this case we use SHA1 for protection. In these figures mentioned above, we add the identification scores for each detection. We use these scores to update our dynamic knowledge base.

VII. Conclusion

To conclude this paper discusses a "PPC" solution (PPC: Performance, Precision, Confidentiality). We presented in this work a secure detection approach. With these methods we guarantee the confidentiality of the data. We also discussed, in this paper, the performance of our solution in execution time. Automatic discovery of personal data is the first step in developing effective data governance strategies.

References

[1] Mihir Bellare, Ran Canetti, and Hugo Krawczyk. Keying hash functions for message authentication. In *Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings*, pages 1–15, 1996.

[2] Mehdi Bentounsi, Edouard Cante, Daniel Coya, Patrice Darmon, Arnaud De Chambourcy, and Gisèle Gnokam. ARIANE : la Gouvernance des Données comme Accélérateur de Conformité au Règlement Général sur la Protection des Données. In *BDA 2019 - 35 ème Conférence sur la Gestion de Données - Principes, Technologies et Applications*, Lyon, France, October 2019.

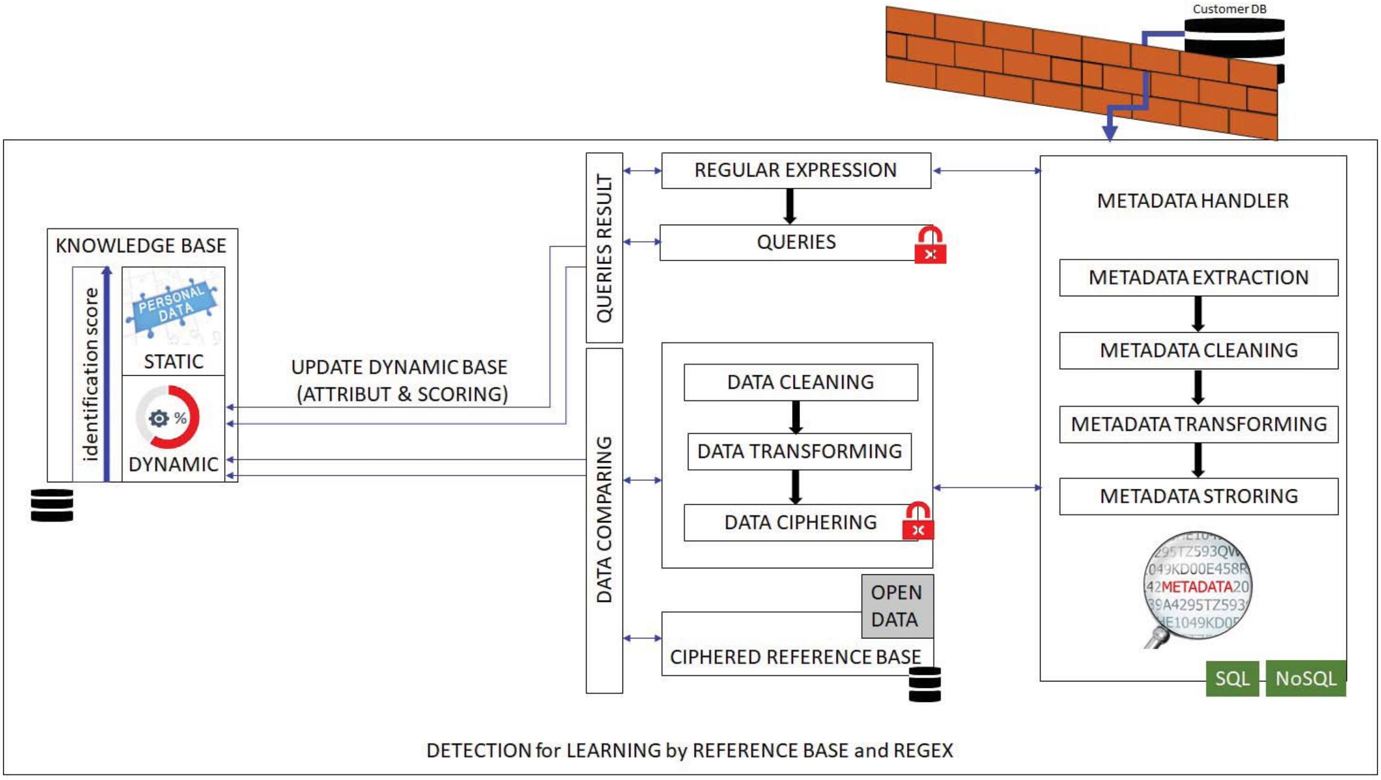

Fig. 2. Architecture for Learning

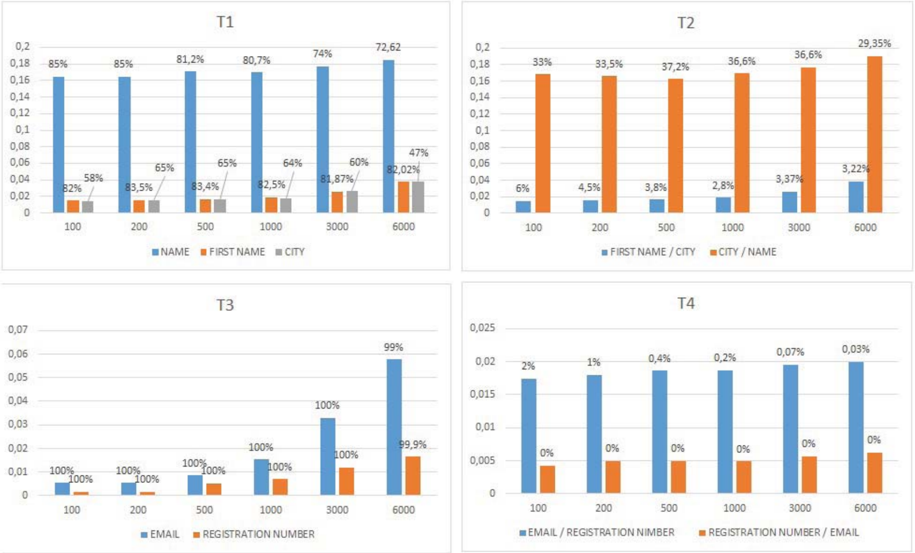

Fig. 4. Performance in SQL MD5

[3] Angela Bonifati, Luigi Palopoli, Domenico Saccà, and Domenico Ursino. Automatic extraction of database scheme semantic properties using knowledge discovery techniques. *Transactions of the SDPS*, 3(1):55–78, 1999.

[4] Wenliang Du and Mikhail J. Atallah. Secure multi-party computation problems and their applications: a review and open problems. In *Proceedings of the New Security Paradigms Workshop 2001, Cloudcroft, New Mexico, USA, September 10-13, 2001*, pages 13–22, 2001.

[5] Michael J. Freedman, Kobbi Nissim, and Benny Pinkas. Efficient private matching and set intersection. In *Advances in Cryptology - EUROCRYPT 2004, International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004, Proceedings*, pages 1–19, 2004.

[6] Otman Manad, Mehdi Bentounsi, and Patrice Darmon. Enhancing talent

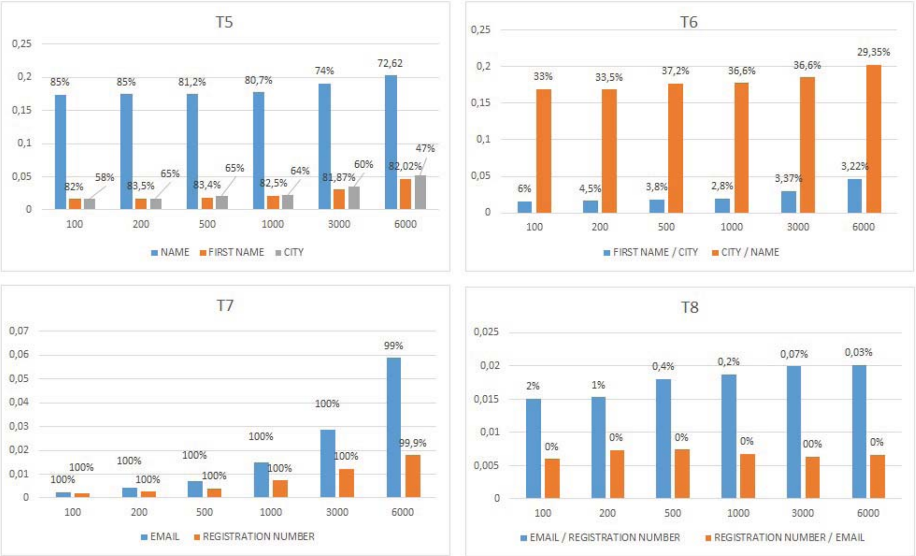

Fig. 5. Performance in SQL with SHA1

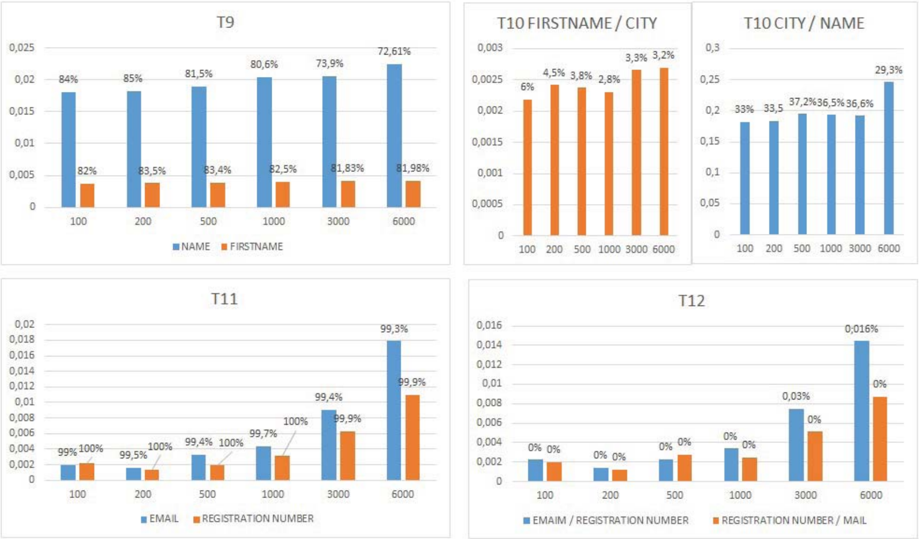

Fig. 6. Performance in MongoDB with SHA1

search by integrating and querying big HR data. In *IEEE International Conference on Big Data, Big Data 2018, Seattle, WA, USA, December 10-13, 2018*, pages 4095–4100, 2018.

[7] Richard Marciano, William Underwood, Mohammad Hanaee, Connor Mullane, Aakanksha Singh, and Zayden Tethong. Automating the detection of personally identifiable information (PII) in japanese-american WWII incarceration camp records. In *IEEE International Conference on Big Data, Big Data 2018, Seattle, WA, USA, December 10-13, 2018*, pages 2725–2732, 2018.

[8] Colin Tankard. What the GDPR means for businesses. *Network Security*, 2016(6):5–8, 2016.

[9] Roland L. Trope, E. Michael Power, Vincent I. Polley, and Bradford C. Morley. A coherent strategy for data security through data governance. *IEEE Security & Privacy*, 5(3):32–39, 2007.

[10] Atsushi Yamada and Michael Peran. Governance framework for enterprise analytics and data. In *2017 IEEE International Conference on Big Data, BigData 2017, Boston, MA, USA, December 11-14, 2017*, pages 3623–3631, 2017.

[11] Andrew Chi-Chih Yao. Protocols for secure computations (extended abstract). In *23rd Annual Symposium on Foundations of Computer Science, Chicago, Illinois, USA, 3-5 November 1982*, pages 160–164, 1982.