A Methodological Approach for Big Data Security: Application for NoSQL Data Stores

Houyem Heni(✉) and Faiez Gargouri

Miracl Laboratory, Higher Institute of Computer Science and Multimedia,
Sfax University, Sfax, Tunisia
houyem.heni@gmail.com, faiez.gargouri@isimsf.rnu.tn

Abstract. Securing big data is among the major challenges for information suppliers. Indeed, the lack of a robust methodological solution dedicated to the big data security makes the issues of privacy and personal data protection major research areas. In fact, many studies and works have dealt with the meeting between privacy and big information. Because of the huge volume of data that spread between social networks and clouds Application, we have to think about an approach that addresses enhancing data security in databases, specifically in the context of NoSQL environments. This paper introduces a new methodological approach for big data security based on data fragmentation.

Keywords: Big data · Security · Sensitive data · NoSQL · Mongo DB · Data fragmentation

1 Introduction

The quantity of heterogeneous data available on the web and in big databases is growing exponentially. According to research conducted by IDC, the volume of digital data will increase from 40 to 50 % every year, reaching a total volume of 40 bytes zeta-2020. As shown in [1], private and governmental organizations are increasingly gathering and maintaining vast amounts of data known as big data which often include sensitive personally identifiable information. Considering the three main characteristic of big data knowing as 3Vs: Volume Variety and velocity, we will focus on the security solution for big data. The privacy of the latter and the identity protection is, now, considered as a very important issue. Consequently, securing big data against some damages and information leakage is a critical goal, that's why the meeting between big data and security represents one of the biggest challenges for digital information suppliers. Because sensitive data is everywhere, the importance of data security has been growing during the last few years. In fact, some big business companies have to start investing in big data security such as IBM[1]. It could enlarge these perspectives on security while implementing 6,000 IBM Security experts worldwide, 3,000 IBM security patents, and 4,000 IBM managed security client services worldwide. Although other survey has also

[1] http://www.ibm.com/systems/data/flash/lv/pdf/venkateshSadayappan_IBMForum2013.pdf.

S. Arik et al. (Eds.): ICONIP 2015, Part IV, LNCS 9492, pp. 685–692, 2015.
DOI: 10.1007/978-3-319-26561-2_80

demonstrated that efficiently securing sensitive data has become an imperative concern. Following [2], a survey on enterprise data security conducted by Independent Oracle Users Group (IOUG) in 2012 shows that only 50 % of the inquired companies increased their investment in IT security. These factors include variables; such as large-scale cloud infrastructures, diversity of data sources and formats as well as streaming nature of data acquisition. Consequently, traditional security mechanisms are tailored to secure small-scale static [3]. This meeting between big data disturbed stores and security represents one of the biggest challenges for suppliers of digital information. The remainder of this paper is organized as follows. Section 2 depicts data security solutions. In Sect. 3, we discuss the related works that deal with different security techniques. Section 4 is devoted to a study that meets the big data and NoSQL. Section 5 represent an overview of our approach. Finally, we end up this paper with a conclusion.

2 A Survey on Big Data Security Techniques

When talking about information security, there are three things to keep in mind: Confidentiality, Integrity, and Availability, or CIA. Moreover As [4], there are other Concepts related to people who use that information. For instance, authentication is proving that a user is the person he or she claims to be. Besides, authorization is the act of determining whether a particular user or computer system has the right to carry out a certain activity [5]. In the following section, we present an overview of the most relevant works proposed in the literature related to the big sensitive information security.

Online big data applications are vulnerable to theft of sensitive information because adversaries can access to private data that can be captured or leaked by curious administrators. In fact, the problem of privacy and sensitive data protection have frequently been one of the major concerns for suppliers of vast and complex information databases during the last few years, especially in the context of social networking. As depicted in [1], the various security matters and upcoming challenges are reviewed in terms of standards; such as PCI-DSS, ITIL, and ISO- 27001/27002. In our approach, we will consider a survey on Big Data security. Security and privacy issues are magnified by the three V's of big data: Velocity, Volume, and Variety. These factors include attributes like large-scale cloud infrastructures, heterogeneous formats, streaming nature of data acquisition and the increasingly big volume of inter-cloud sharing[2]. As the Oracle big data appliance known as OBDA the Database, customers have benefited from a rich set of security features: encryption, redaction, data masking, database firewall and label-based access control. Oracle wants similar capabilities with their Hadoop cluster. In [7] the OBDA believe that an essential secret of an optimized appliance is protecting its data. Therefore, by default, the BDA delivers the "AAA of security": authentication, authorization and auditing for Hadoop cluster which represent the most important element in big data environment. Oracle, also, adds encryption of data-at-rest on Big Data Appliance. This encryption can be done in two modes. The first one leverages the

[2] https://cloudsecurityalliance.org/media/news/csareleases-the-expanded-top-ten-big-data-security-privacychallenges/.

Trusted Platform Module (TPM) on the motherboard to provide a key to encrypt the data on disk. This mode does not require a password or pass phrase but relies on the motherboard. The second mode leverages a passphrase, which in turn will be used to generate a private-public key pair generated with Open SSL.

3 Related Works

We notice that the harvesting of big data sets and the use of cloud applications imply security concerns. The tasks of ensuring security for sensitive and personal data and the privacy protecting at an age where big data has been transmitted and shared with high speed around the world, have become harder. Lots of research works have been done to treat these mass data security. In [8], the authors enumerate many benefits of Privacy in area of big data. They believe that Privacy and data protection laws are premised on individual control over information and on principles; such as data minimization and purpose limitation. However, they did not give a perfect solution for big data security. Given that the cloud is rich in big data thanks to its applications and distributed layers and it is important to know the security solution used in this dematerialized world as shown in [9], the authors proposed a combined approach which provides a way to protect the data and check the integrity and authentication by following the best possible industry mechanisms. This technique introduces the fragmentation of data into different sections: Index builder, 128-bit SSL encryption, message authenticate code and a double authentication of user; one by owner and the other by cloud and verification of the owner's digital signature. It, also, provides availability of data by surpassing many issues like data leakage, tampering of data and data encryption. In fact, data fragmentation was applied as a perfect solution to more ensure security in data bases. According to [1], fragmentation and encryption provide protection of data in storage. The authors address these issues by proposing a solution to enforce data collections privacy that combines data fragmentation with encryption. They model privacy requirements as Confidentiality constraints expressing the sensitivity of attributes and their associations. Then, they use encryption as an underlying measure for making data unintelligible, while exploiting fragmentation as a way to break sensitive associations among attributes. In [10], they introduced the problem of privacy-aware data partitioning; namely the problem of splitting a sensitive dataset amongst untrusted parties. In their work, the authors present SPARSI, a theoretical framework that allows us to formally define the problem, as an optimization of the tradeoff between the utility derived by publishing the data and the maximum information disclosure incurred to any single adversary.

4 Big Data and NoSQL

The emergence of cloud computing and its distributed applications creates the need for huge bases to store the large amounts of structured, semi-structured and unstructured data. When we talk about big data and its storage we think about NoSQL data bases.

NoSQL[3] is not necessarily to be taken as “not”, but “not only” SQL. It offers solutions that can complement conventional RDBMS solutions. Non-relational data stores have not yet reached security infrastructural maturity. NoSQL Databases were built to tackle different obstacles brought about by the analytics world; hence security was never part of the model at any point of its design stage. According to [6] Developers using NoSQL databases usually embed security in the middleware. Furthermore, NoSQL databases do not provide any support for enforcing it explicitly in the database. However, clustering aspect of NoSQL databases poses additional challenges to the robustness of such security practices. In fact, there are four types of NoSQL databases. In [11] the authors draw a comparison between these databases as follows:

- Key/value stores: store items as alpha-numeric identifiers (keys) and associate values in simple, standalone tables. Requests can only be performed against keys, not values.
- *Column stores*: do not store data in tables but in massively-distributed architectures. In column stores, each key is associated with one or more attributes (columns). The data stored is based on the sort order of the column family.
- *Document data bases*: are designed to manage and store documents encoded in a standard data exchange format; such as XML. Unlike the simple *key-value stores*, the value column contains semi-structured data, specifically attribute/value pairs. The number and type of recorded attributes can vary from row to row. Both keys and values are searchable. Document databases are good for storing and managing big data-size collections of literal documents like text documents, email messages, and XML documents.
- *Graph databases*: replace relational tables with structured relational graphs of interconnected key value pairings. The graphs are represented as an object-oriented network of conceptual objects (nodes), relationships (edges) and properties (object attributes expressed as key-value pairs).

We frame our work in the context of NoSQL databases. They are by far the most common solution for the management of the data privacy subject We are, also, convinced that document oriented databases are the suitable ones, because they are Scalable, schema-less and very flexible. Besides, these databases are intended to Store structured, semi-structured and unstructured data.

5 Overview of the Approach

As we mentioned above, one of the most important challenges of big data security is first the protection of sensitive data management and second how to determine appropriate methods and algorithms to ensure security. Big data has a disturbed technology. In fact, the data in big data environment has to go through different processing levels. Thus, the security mechanism should be efficient and provided at each step. Data should not succumb to the attackers trying to retrieve or tamper with it and even the big data provider should not be able to harm the data in any possible manner, because big data

[3] http://searchdatamanagement.techtarget.com/definition/NoSQL-Not-Only-SQL.

suppliers cannot be trusted with data of high sensitivity. For this reason, it is obvious that the proposed approach has been designed by keeping all these things in mind. Besides, in comparison to prior works, it provides all these required measures to protect data in a very efficient and organized manner. This approach will be applied to MongoDB. In order to explain this approach, Fig. 1 describes this method which consists of four phases:

1. Pre-processing phase: this phase is use to group automatically big data in NoSQL database.
2. Search and identify sensitive data phase: In this step, we can pass by a Learning Algorithms based on neural networks to identify sensitive data
3. Data Fragmentation phase: it is used to provide a very high security. We think about Data Fragmentation to better protect the data that has a sensitive and personal aspect. In this step, we must propose our algorithm based on data division followed by an encryption stage for the fragmented part.
4. Data Reconstruction phase

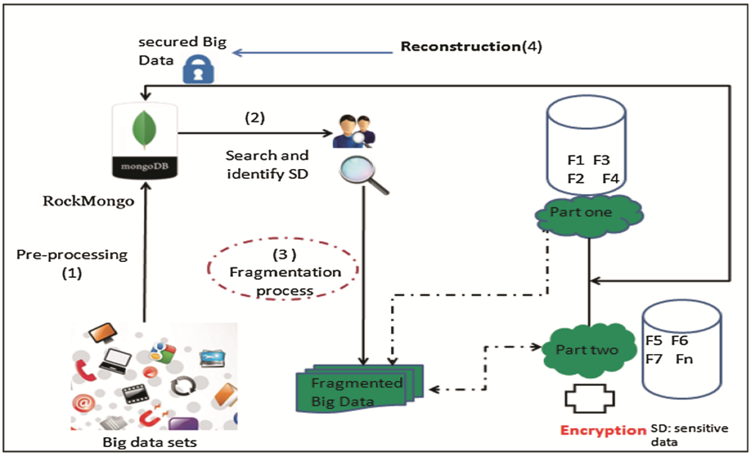

Fig. 1. Data security scenario based on data fragmentation

5.1 The Pre-treatment Phase

This step represents our preprocessing works. The main goal is to group automatically Big Data (structured, semi- structured and unstructured data) in a NoSQL database. Then, we feed MongoDB with heterogeneous data because it is known for its simplicity and high performance in processing big amounts of data. Since it is distributed in treatment, the MongoDB will facilitate the work of fragmentation. In this pre-processing step, we choose the Rock mongo[4] which is a MongoDB management tool, written in

[4] http://docs.mongodb.org/ecosystem/tools/administration-interfaces/

PHP 5 and dedicated for administering a mongo database to feed our data base and to insert heterogeneous data and save as json format in our document-oriented database.

5.2 Find and Identify Sensitive Attributes Phase

This phase deals with mechanisms and methods of classifying and identifying the sensitive attributes sets from collections stored in MongoDB. In many studies, the authors focus their work on attributes sets. For instance, in [1, 12], they worked on attributes sets and they split them into fragmentations sets f1…fn. In our approach, we will split the attribute itself after being fragmented.

During this step, we will categorize our data collections, i.e. group attributes of class from the collections. Then, we propose clustering algorithm. The used algorithm is a neural network which takes as input the MongoDB collection sets and is given as output many categories so each attributes will be classified in an appropriate category which allows two classes known as sensitive data and ordinary data. After this process, we will go through identification to highlight the difference between ordinary attributes and sensitive attributes. In the literature, there are many software used to discover sensitive data residing whatever in computers or in the massive parallel bases or even in the clouds. But, we believe that inventing a learning algorithm will better serve our request, especially when it comes to a huge volume of heterogeneous data. This learning algorithm aims at identifying sensitive data among ordinary information, i.e. if it comes from a succession of figures or address mail or secret code.

5.3 Sensitive Data Fragmentation Phase

This phase is summarizing the scenario of big data security. In many large-scale solutions, data is divided into separate partitions that can be managed and accessed separately. The partitioning strategy must be chosen carefully to maximize the benefits while minimizing adverse effects.This phase is further divided into two sub-sections.

Data Fragmentation

After identifying and searching sensitive data, each sensitive attribute must be split in two parts. As shown in [13], Data fragmentation is a process of division or mapping database where the database is broken down into a specific number of parts. Then, it is stored in the site or units of different computers in a data network. When applying fragmentation, data must meet several conditions to obtain optimal fragmentation. Below we illustrate some data fragmentation principles:

Completeness: a unit of data that is still in the main part of the relationship. Then, the data must be in one fragment. When there is a relation, the distribution of the data must be an integral part of the relationship.

Reconstruction: an original relation can be reused or combined return of a fragment. When it breaks down, data is still possible to be combined again with no change in the structure of data.

Disjointness: data within the fragment should not be included in the other fragments in order to avoid redundancy of data, except for primary key attributes of vertical fragmentation. Fragmenting means splitting sets of attributes so that they are not visible together; that is to say, the associations among their values are not available without access to the encryption key.

Data Encryption

As we mentioned above, to better ensure the security of sensitive attributes and after having split it in two parts, we will save the second part in somewhere else to prevent the hackers to find the full information. Thus, we think of a classic approach among the security algorithms. We choose encryption as a perfect solution. As [14, 15] demonstrate, the encryption is the best method to protect sensitive data at the database level while maintaining high database performance. Encryption is applied after the fragmentation sets level. Finally, the encryption key is given to the authorized users needing to access the information. Users that do not know the encryption key as well as the storing server are able neither to access sensitive information nor to reconstruct the sensitive associations. To better ensure security, we will apply an encryption algorithm after the Fragmentation process.

5.4 Data Reconstruction

Relation between sensitive attributes after fragmentation can be reused. When it breaks down, data is still possible to be combined again with no change in the data structure. It is worth-noting that in data reconstruction, there are many constraints; such as choosing the perfect fragment data to the adequate query.

6 Conclusion

In the early 21st century that witnessed the explosion of Big Data, the security of data with sensitive content is still considered as a very important and effective axis of research. This paper presents a big data protection scenario. The objective of this approach is to overcome the threads and leaks in big data environment, especially for NOSQL data bases. In this context, we use the data fragmentation combined with a data Encryption applied in MongoDB.

As future works, we envisage to automatically feed NoSQL database, to define Confidentiality constraint for NoSQL data bases, to implement learning algorithm, to identify sensitive attributes and to apply a fragmentation process appropriate to MongoDB collections.

References

1. Ciliani, V., De Capitani, S., Vimercati, D., Foresti, S.: Combining fragmentation and encryption to protect privacy in data storage. ACM **13**(3), 1–30 (2010)

2. McKendrick, J., IOUG Enterprise Data Security Survey: Closing the Security Gap, the Independent Oracle Users Group (IOUG) Security report, November 2012
3. Cloud security appliance, the-expanded-top ten-big-data-security-privacy challenges, CSA White Paper (2010)
4. dos Santos, R.J.R.: enhancing data security in data warehousing, Ph.D. thesis in Information Sciences and Technology, supervised by Professor Jorge Bernardino and Professor Marco Vieira, The University of Coimbra, February 2014
5. Pesante, L.: Introduction to Information Security. Carnegie Mellon University, Pittsburgh (2008)
6. Zvarevash, K., Mutandavari, M., Gotora, T.: A survey of the security use cases in big data. Int. J. Innov. Res. Comput. Commun. Eng. **2**(5), 4259–4266 (2014)
7. Oracle Corporation: New Big Data Appliance Security Features. Oracle White Paper, November, 2013
8. Tene, O., Polonetsky, J.: Privacy in the age of big data: a time for big decisions, 64 Stan. L. Rev. Online 63, 2 February 2012
9. Sood, K.S.: A combined approach to ensure data security in cloud computing. J. Netw. Comput. Appl. **35**(6), 1831–2012 (2012)
10. Rekatsinas, T., Deshpande, A., Machanavajjhala, A.: A SPARSI: partitioning sensitive data amongst multiple adversaries. PVLDB **6**(13), 1594–1605 (2013)
11. Moniruzzaman, A.B.M., Hossain, S.A.: NOSQL database: new era of databases for big data analytics- classification, characteristics and comparison. Int. J. Database Theory Appl. 6(4) (2013)
12. Vinogradov, S., yak, A.P.: Evaluation of data anonymization tools. In: The Fourth International Conference on Advances in Databases, Knowledge, and Data Applications (2012)
13. Navaz, S.: A.S., Prabhadevi, C., Sangeetha, V.: Data grid concepts for data security in distributed computing (0975 – 8887). Int. J. Comput. Appl. **61**(13), 6–11 (2013)
14. Oracle Corporation, "Security and the Data Warehouse", Oracle White Paper, April 2005
15. Oracle Corporation, Oracle Advanced Security Transparent Data Encryption Best Practices, Oracle White Paper, July 2010