Big Data – Privacy and Management

Sanjay Thimmarayappa
Department of Information Science and Engineering,
M.S. Ramaiah Institute of Technology, Bangalore

Megha V
Department of Computer Science and Engineering,
Acharya Institute of Technology, Bangalore

ABSTRACT

With big data dimensions increasing exponentially, the privacy and security issues surrounding it also magnify. Therefore, traditional security mechanisms tailored to secure small-scale, static data are insufficient. Along with the security concerns, data management also becomes an important factor to be considered. This paper discusses about big data and the challenges faced by big data with regard to privacy and management while proposing possible alternatives to overcome these challenges.

Keywords

Big Data, Management, Privacy.

1. INTRODUCTION

Big data is a term that refers to massive amounts of digital information. Data generated from a myriad of sources contribute to this humongous heap of valuable information. This huge amount of structured and unstructured data cannot be processed effectively using traditional databases and software technologies and the Big Data analytics rectify this drawback to a great extent. Three terms outline big data, they are:

i. Volume: This term refers to the amount of data being collected. The various factors contributing towards increasing volume include storing transaction data, live streaming data, data from various sensors, etc.

ii. Variety: Data can be stored in various formats, for example database, csv, access or even in text format.

iii. Velocity: This term refers to the speed at which data is being generated and how fast it needs to be processed to meet the demands. Data movement now is almost real time and the update window has reduced to fractions of seconds.

There are two more dimensions that need to be considered with respect to big data. These provide much more information regarding the data collected.

iv. Variability: The frequency with which the data flows can be highly inconsistent. Periodic peaks tend to overwhelm the system, and we should be well equipped to face such a situation.

v. Complexity: Data coming from multiple sources needs to be linked, matched, cleansed and transformed into certain formats before actual processing begins.

Large scale distributed applications that usually work with large data sets form the big data application domain. On such a huge scale, traditional database applications find it hard to effectively analyse and explore the input data. This problem paves way for the development of Big Data applications. Currently, Google's Map Reduce framework and the Apache Hadoop are the most important software systems for big data applications. These software systems generate a vast amount of intermediate data. Some of the famous Big Data use cases include:

i. A complete view of customer preferences- The huge amount of click stream data generated by users on a retail website is useful to the retailer to determine customer preferences. This unstructured data after being processed and converted into structured data along with the help of social media sentiment can provide the retailer the desired view of the customer profile.

ii. Internets of Things - There are a lot of gadgets/sensors that generate valuable information from the customers' environment regarding their health, security etc.

iii. Data warehouse optimization - Due to the voluminous size of enterprise data warehouse, it is vital to store unstructured archive data in a cost effective Hadoop platform.

iv. Information security - To store the huge amounts of machine or event data generated, it is imperative to replace traditional systems like relational databases with Hadoop which is a much more efficient alternative.

1.1 Need for Security and Data Management

Big data being a new technology has the ability to introduce new vulnerabilities if it is not well understood by the companies using it. Since most of the data in the data sets in valuable, securing it from a wide array of security breaches becomes a top priority. Minimizing the effects of a breach is also very important.

Many businesses use big data for marketing and research purposes. But, these businesses may not have the fundamental assets to secure the data from a security perspective. Now that the companies are storing sensitive user information in their data sets, a security breach might pave way for serious legal repercussions and reputational damage. In this new era, companies are using technology to store and analyze enormous amounts of data about their company, business, customers etc. As a result, information classification becomes a critical aspect. Employing techniques like encryption, logging, honeypot detection is necessary to make big data secure.

As the variety of file formats increases, managing big data becomes a tedious task. Big data management tends to this challenge and makes sure that there is a high level of data quality and accessibility for data analytics operations. With companies having huge data sets relating to their customers, it is vital to make sure only the useful information is retained. Otherwise, in case of a security breach the situation could worsen. Employing auto tiering strategies is an option which has been discussed in the following paragraphs. Removing the unwanted information is also important to reduce data set size which can be analyzed quickly and productively.

2. DATA MANAGEMENT

Multi-tiered storage media which stores data and transaction logs is hard to operate on as the size of data sets continue to grow exponentially. The IT manager manually controls the movement of the data between the layers and has control over what data is moved and when it is moved. Hence, auto tiering solutions were introduced. Without the ability to keep track of data storage, auto tiering poses new challenges to secure data storage. New mechanisms need to be developed to avoid unauthorized access and to maintain constant availability.

2.1 Use Case

Consider a situation where a company needs to integrate data from different divisions. This data could include information that is constantly used along with information that is accessed infrequently, like the research and development results. The data that is least accessed is stored in the lower tiers by the auto-tier storage system and since the lower–tier often provides lower security than the higher tiers, the information in the lower tiers is at a higher risk of security breach. Hence, the company should carefully study tiering strategies. Apart from these risks, perpetrators could also induce data inconsistency and disputes among users by accessing the meta-data i.e.; text logs and this too needs to be countered.

2.2 Modelling

Even though auto-tier storage system is a promising solution, it generates risks due to untrusted storage service or inconsistent security policies. This system is a transparent service with good scalability and elasticity. However, this system still possesses a threat which includes seven major scenarios:

i. Confidentiality and integrity: The storage service providers are assumed to be untrustworthy third parties alongside those attempting to steal information. Activity log of users and the data transmissions among the tiers provide clues regarding user activities. With the help of these clues, the service providers have the ability to decrypt information, and certain characteristics can be retrieved.

ii. Provenance: It is very difficult to access large data sets for their availability and integrity. Hence, lightweight schemes are employed to probabilistically check for its verification. This method undergoes low communication and computing overhead.

iii. Availability: There is performance gap between the lower tiers and the higher tiers, which puts the lower tiers at higher risk of being affected by attacks like the denial of service (DOS). Therefore, the service providers must guarantee constant availability so that the recovery process from such attacks is fast.

iv. Consistency: Data is used by multiple users and hence consistency must be maintained among multiple duplicates of the data which is stored in multiple locations. The two issues that must be addressed in this regard are:

   a. Write serializability.
   b. Multi-writer and multi-reader problem.

v. Collusion attacks: The data owner stores the cipher text in the auto tier storage system and only desired users are given the key and access permission to certain parts of data. Even the service provider needs the key to access the data and even if the service provider shares the key among the users, they must not be able retrieve the correct information.

vi. Roll back attacks: “user’s freshness” must be checked. This refers to a scenario in which the service provider chooses to roll back the changes made to the data and instead deliver an out-dated version of the information to the user. Certain methodologies must be employed by the user to check if the data is updated.

vii. Disputes: Access log of the user must be maintained. This will prevent disputes between storage service provider and the user over outsourced data.

2.3 Analysis

With the recent advancements in security fields, confidentiality and integrity can be achieved by employing techniques like message-digests and robust encryption. Exchanging signed message digests is an option to address potential disputes [4]. Persistent Authenticated Dictionary [1] (PAD) and chain hash [7] or periodic audit [8] can be used to solve user’s freshness and the write-serializability problem. Linear and concurrent lock-free protocols can be used to solve single write and multi-read problem [6]. There are multiple techniques to handle provenance issues. Key rotation [5] and Broad encryption [3] can be used to solve scalability. As long as users do not exchange their private keys, policy based encryption system (PBES) can guarantee a collusion free environment [2]. If exchange of private key is required, then mediated decryption system can avoid collusion attack. Even though techniques for every problem in large scale auto-tier system are present, securing inter tier data transmissions is a challenge due to the non-uniform security policies that different tiers adopt. More considerations are necessary to balance security, usability, complexity and cost.

2.4 Implementation

With many security strategies in multi-tier storage systems, various structures can be employed to meet the general security requirements. Among them, three special cases require more attention:

i. Dynamic data operations- Since operations such as modification, deletion, duplication and insertion are performed frequently on the data sets in an auto tier system, the data sets are dynamic. The extended version of PDP scheme relies only on symmetric-key cryptography, through which higher efficiency can be achieved. However, it cannot support fully dynamic data operations.

ii. Privacy preservation- A public auditing scheme with focus on privacy-preserving was proposed for storage in Wang [9] based on homomorphic linear authenticator integrated with random masking. This scheme was able to preserve data privacy even when the TPA audits data sets that were stored in servers at different tiers.

iii. Secure manipulations- Encrypted cipher text can be operated upon by employing a fully homomorphic encryption scheme [12]. Establishing a reliable IAAS storage is an alternative. This reliable structure needs to be constructed on top of the untrusted infrastructure.

3. DATA PRIVACY

A recent analysis shows how companies leverage data analytics for marketing purpose. One such instance was when AOL released anonymized search logs for academic purposes, but these logs were used to identify users easily based on their searches [10]; a similar situation was faced by Netflix users when they were identified by correlating their Netflix movie scores with IMDB scores [11]. As a result, data for analytics is not enough to maintain user privacy and hence guidelines and recommendations for preventing accidental privacy disclosures are important.

3.1 Use Case

Large organizations collect user data and this data is constantly being accessed by inside analysts, outside contractors and also business partners. An untrusted partner or malicious insider can modify and extract private information from customers. Also, intelligence agencies require large amount of data which include chat-room messages, personal blogs and network routers. Since most of this data is innocent, we can preserve anonymity by making sure that none of this data is retained. Everyone's safety is increased by applying the concepts of robust and scalable privacy-preserving mining algorithms.

3.2 Modelling

In big data stores, there are multiple ways in which user privacy is compromised. The company hosting the data can have a lot of vulnerabilities which be exploited by a malicious attacker. A threat model for privacy of user data shows three important scenarios:

i. An insider in the company hosting the big data store can abuse his/her level of access and violate privacy policies. For example, a Google employee stalked teenagers by monitoring their Google chat communications [13].

ii. If data is outsourced for analytics by the data owner, an untrusted partner could misuse their access to the data to deduce private information from users. This is similar to the utilization of big data in the cloud computing model where the cloud infrastructure is not usually controlled by the data owners.

iii. Sharing data for research is important. However, with re-identification techniques i.e., matching anonymized data with its true owner, ensuring full anonymity of the data is a challenge.

3.3 Analysis

One of the best way protect user privacy is to continuously monitor user data. Privacy-preserving analytics being an open area of research can curb the success of malicious actors from data sets. There are a few practical solutions but differential privacy is a good first step towards privacy preservation. It allows us to reason formally about what an adversary could learn from released data, while avoiding the need for many assumptions, the failure of which has led to many privacy violations in the past [14]. But this technique needs an advanced solution to address the issues like the computation overhead associated with it. Another potential solution would be universal homomorphic encryption. This promises to provide data analytics on an encrypted version of the outsourced data. This technology is currently not practical for deployments. Leakage of private information is another concern and it needs to be controlled under composition i.e., when multiple databases are linked. Linking anonymized data is also a challenge since we need to maintain consistency to ensure data compatibility among different environments.

3.4 Implementation

Some basic steps to avert attacks from outsiders with malicious intents include encryption of data, access control, and authorization mechanism. Software infrastructure must be patched up with the latest security solutions.

Big data operators can prevent insider attacks by employing the separation of duty principles, which force malicious insiders to collude. Also, a strict policy to keep track of logging access to data sets can assist forensics and act as a deterrent, notifying probable malicious intruders that their activities can be traced. Data sharing is currently an open area of research and a best practice recommendation for this is to be aware of re-identification techniques.

4. CONCLUSION

With every passing day, the computing environment is becoming more and more affordable. This leads to situations where system and analytics environments are shared over cloud and application environments become heavily networked.

This paper highlights the privacy and management issues that need to be addressed to make the big data processing and computing infrastructure more secure and manageable. Effective alternatives for various scenarios have been proposed.

Big data operators can prevent insider attacks by employing the separation of duty principles. Logging access to data sets acts as a deterrent to possible malicious intruders from using data they were not entitled to. By applying the principles of dynamic data operation, data preservations and secure manipulations, data can be effectively managed.

We hope that this paper will spawn more research and development on this topic and inspire communities to collaboratively focus on the barriers to greater security and privacy in big data platforms.

5. REFERENCES

[1] Anagnostopoulos, Aris, Michael T. Goodrich, and Roberto Tamassia. "Persistent authenticated dictionaries and their applications." *Information Security*. Springer Berlin Heidelberg, 2001. 379-393.

[2] Bagga, Walid, and Refik Molva. "Collusion-free policy-based encryption."*Information Security*. Springer Berlin Heidelberg, 2006. 233-245.

[3] Boneh, Dan, Craig Gentry, and Brent Waters. "Collusion resistant broadcast encryption with short ciphertexts and private keys." *Advances in Cryptology–CRYPTO 2005*. Springer Berlin Heidelberg, 2005.

[4] Feng, Jun, Yu Chen, and Pu Liu. "Bridging the missing link of cloud data storage security in AWS." *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*. IEEE, 2010.

[5] Kallahalla, Mahesh, et al. "Plutus: Scalable Secure File Sharing on Untrusted Storage." *Fast*. Vol. 3. 2003.

[6] Majuntke, Matthias, et al. "Abortable fork-linearizable storage." *Principles of Distributed Systems*. Springer Berlin Heidelberg, 2009. 255-269.

[7] Onieva, Jose A., and Jianying Zhou. *Secure multi-party non-repudiation protocols and applications*. Vol. 43. Springer, 2008.

[8] Popa, Raluca Ada, et al. "Enabling Security in Cloud Storage SLAs with CloudProof." *USENIX Annual Technical Conference*. Vol. 242. 2011.

[9] Wang, Cong, et al. "Privacy-preserving public auditing for data storage security in cloud computing." *INFOCOM, 2010 Proceedings IEEE*. Ieee, 2010.

[10] Duhigg, Charles. "How companies learn your secrets." *New* York *Times* 16 (2012).

[11] Barbaro, Michael, Tom Zeller, and Saul Hansell. "A face is exposed for AOL searcher no. 4417749." *New York Times* 9.2008 (2006): 8For.

[12] Gentry, Craig. "Computing arbitrary functions of encrypted data."*Communications of the ACM* 53.3 (2010): 97-105.

[13] A.Hough. "Google engineer fired for privacy breach after 'stalking and harassing teenagers'". The Telegraph. Sept 15, 2010.

[14] Haeberlen, Andreas, Benjamin C. Pierce, and Arjun Narayan. "Differential Privacy Under Fire." USENIX Security Symposium. 2011.