IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 5, Ver. III (Sep. - Oct. 2016), PP 68-70
www.iosrjournals.org

Design Approach to Big data Systems in Developing and Maintaining the Information Security Systems

Dr.Mani Sarma Vittapu
Assistant Professor, Centre of ITSC, Addis Ababa Institute of Technology, Addis Ababa University, Addis Ababa, Ethiopia.

***Abstract**: Data is accumulating from almost all aspects of our everyday lives that it becomes huge and multi-structured and has hidden useful information. The challenges with Big Data include capture, curation, storage, search, sharing, transfer, analysis, and visualization. Big Data provides materials for mining hidden patterns to support innovation mostly by data mining. The interaction research with Big Data support methods for innovation is rare at present. Knowledge discovered by data mining is novel and quantitative. However, it still lacks a uniform knowledge management model to support the innovation process effectively. The fact shows that since there emergence, big data techniques are changing very fast. In this paper, developing and maintaining the information security system to illustrate how those big data systems evolve and also describes some key design factors and challenges for future big data systems. Proposed design generic systems that can provide near real-time analytic services for many netease applications, such as spam detection, game log analysis and social community mining. No solutions can address all big data problems, especially when data size keeps increasing, more complex user requirements need to handle, the emergence of new hardware violates the old design and the old system becomes too complicated for maintenance. We face a series of technical challenges that have not been well addressed by both academic community and industry.*

***Keywords:** Map Reduce, Real time analysis, Information Security.*

I. Introduction

Nowadays the Internet represents a big space where great amounts of information are added every day. The IBM Big Data Flood Info graphic shows that 2.7 Zettabytes of data exist in the digital universe today. Also according to this study there are 100 Terabytes updated daily through Face book, and a lot of activity on social networks this leading to an estimate of 35 Zettabytes of data generated annually by 2020. Just to have an idea of the amount of data being generated, one zettabyte (ZB)
equals 10^{21} bytes, meaning 10^{12} GB. We can associate the importance of Big Data and Big Data Analysis with the society that we live in. Today we are living in an Informational Society and we are moving towards a Knowledge Based Society. The same happens with Big Data. Every organization needs to collect a large set of data in order to support its decision and extract correlations through data analysis as a basis for decisions.

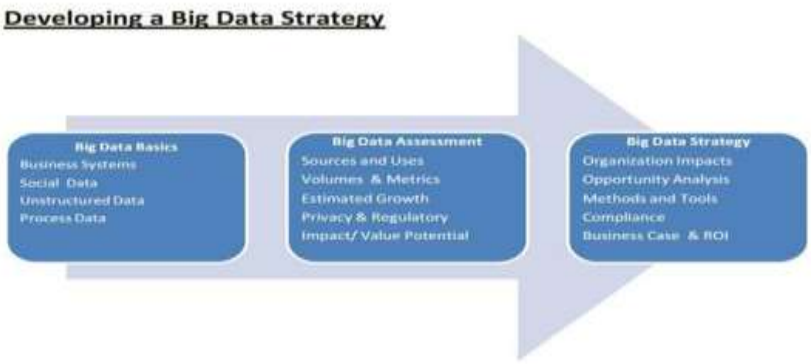

Fig 1. Developing a Big Data Strategy

The understanding of Big Data is mainly very important. In order to determine the best strategy for a company it is essential that the data that you are counting on must be properly analyzed. Also the time span of this analysis is important because some of them need to be performed very frequent in order to determine fast any change in the business environment. Another aspect is represented by the new technologies that are developed every day.

2. Threats of Big Data Security

Just as Gartner said: "big data information security is a necessary fight"[3]. Today, big data has penetrated into various industries, and has become a kind of production factor which plays an important role. In the future it would be the highest point of the competition. With the development of rapid processing and analysis technology, the potential information it contained can quickly capture the valuable information in order to provide reference for decision making. However, as big data setting off a wave of productivity and consumer surplus, the challenge of information security is coming either.

2.1 Data Acquisition

The source of big data is diversity. Therefore, the first step to process big data is to collect data from source and pre-process, in order to provide uniform high quality data set to the subsequent process. As a result, due to the inundation of data acquisition, large data become more likely to be "discovered" as a sensitive target, and be more and more attention. On one hand, big data not only means the huge amounts of data, but also means more complex and more sensitive data. These data would attract more potential attackers, and become a more attractive target. On the other hand, with data assembled, the hacker could get more data in one successful attack, and reduce hacker's attack costs.

The confidentiality of information refers that according to a specified requirements, information can not be disclosed to unauthorized individuals, entities or processes, or provided the characteristics of its use. A large amount of data collection includes a large number of enterprises operating data, customer information, personal privacy and all kinds of behavior records. The centralized storage of these data increases the risk of data leakage, and not abused of these data also becomes a part of the personal safety. There is no clear definition to the proprietorship and right to use of sensitive data. And many analysis based on large data did not consider the individual privacy issues involved either.

The integrity of information refers to all the resources which can only be modified by authorized people or with the form of authorization. The purpose is to prevent information from being modified with unauthorized users. Due to the openness of big data, in the process of network transmission, information would be damaged, such as hackers intercepted, interruption, tampering and forgery. Encryption technology has solved the data confidentiality requirements as well as protecting data integrity. But encryption cannot solve all of the safety problems.

2.2 Storage of Data

The formation of network society creates the platform and channel of resource sharing and data exchange for the big data in the field of various industries. Network society based on cloud computation provides an open environment for big data. Network access and data flow provides the basis of rapid elasticity push of the resources and the personalized service. In recent years, from the chain reaction of user account information being stolen on the Internet, it can be seen that big data is more likely to attract hackers, and once being attacked, the volume of stolen data is huge.

Before big data, data storage is divided into relational database and file server. And in current big data, diversity of data type makes us unprepared. For more than 80% of the unstructured data, NoSQL has the advantages of scalability and availability and provides a preliminary solution for big data storage. But NoSQL still exist the following problems: one is that relative to the strict access control and privacy management of SQL technology; Secondly, although NoSQL software gain experience from the traditional data storage, NoSQL still exist all kinds of leak.

2.4 Data Mining

With the development of computer network technology and artificial intelligence, network equipment and data mining application system is more and more widely used, to provide convenient for big data automatic efficient collecting and intelligent dynamic analysis. On the one hand, big data itself exits leak. Big data itself can be a carrier of sustainable attack. Viruses and malicious software code hidden in large data is hard to find. On the other hand, the technique of attack improves. At the same time of the big data technology such as data mining and data analysis gaining value information, the attacker using these big data technology either, just as the two following aspects.

3. Data Security Protection Technique

Key technologies in Security protection fields are in great demands to face the security challenges. In this section, we introduce important relevant fields are **Individual User, Internet Enterprise and Cloud Service Provider.**

3.1 Individual User

As with individual users' information in big data environment, the core and basic techniques to provide privacy protection are still in developing period. Take typical K-anonymity scheme as an example, its early version and optimized version divide quasi-identifiers into groups through tuple generalization and restraining method. When an equivalence class has identical value on some sensitive attribute, attackers are able to confirm its value. In response to this issue, researchers proposed 1-diversity anonymity.

Current edge anonymity schemes are mainly based on adding and deleting of the edges. Edge anonymity can be effectively achieved by adding, deleting and exchanging edges randomly. There are problems in such methods that noises randomly added are exiguity, and protections to anonymous edges are insufficient. An important method is to perform division and aggregation operations to super nodes such as node aggregation based anonymous method, genetic arithmetic based method and simulated annealing method based method.

3.2 Internet Enterprise

Information security is critical important for Internet enterprises. System security adopts techniques such as redundancy, network separation, access control, authentication and encryption [18]. Security issues are caused by openness, boundless, freedom of the networks, the key to solve such issues are making network free from them and turning network into controllable, manageable inner system. As network system is the foundation of application system, network security becomes principal issue. Ways to solve network security issues are network redundancy, system separation and access control

3.3 Cloud Service Provider

CSPs provide following measures to prevent security issues in cloud environment. In order to prevent CSPs from peeping users' data and program, separating power and hierarchical management are needed to control access to data in cloud. Provide different authority in accessing data to service provider and enterprise to ensure data security. Enterprise should have total authority and limit authority to CSP.

In cloud computing environment data separation mechanism prevents illegal access to data, however, we should take care of data leakage from CSPs. Mature techniques as symmetrical encryption, public key encryption are available to encrypt data and then upload data to cloud environment. In cloud environment data division is often used with data encryption i.e. encrypted data are scattered in user end and spread in several different clouds. In the way, any CSP is not able to gain complete data.

4. In - Stream Big Data Processing

The shortcomings and drawbacks of batch-oriented data processing were widely recognized by the Big Data community quite a long time ago. It became clear that real-time query processing and in-stream processing is the immediate need in many practical applications. In recent years, this idea got a lot of traction and a whole bunch of solutions like Twitter's Storm, Yahoo's S4, Cloudera's Impala, Apache Spark, and Apache Tez appeared and joined the army of Big Data and NoSQL systems. This article is an effort to explore techniques used by developers of in-stream data processing systems, trace the connections of these techniques to massive batch processing and OLTP/OLAP databases, and discuss how one unified query engine can support in-stream, batch, and OLAP processing at the same time. At Grid Dynamics, we recently faced a necessity to build an in-stream data processing system that aimed to crunch about 8 billion events daily providing fault-tolerance and strict transnationality i.e. none of these events can be lost or duplicated. This system has been designed to supplement and succeed the existing Hadoop-based system that had too high latency of data processing and too high maintenance costs. The requirements and the system itself were so generic and typical that we describe it below as a canonical model, just like an abstract problem statement.

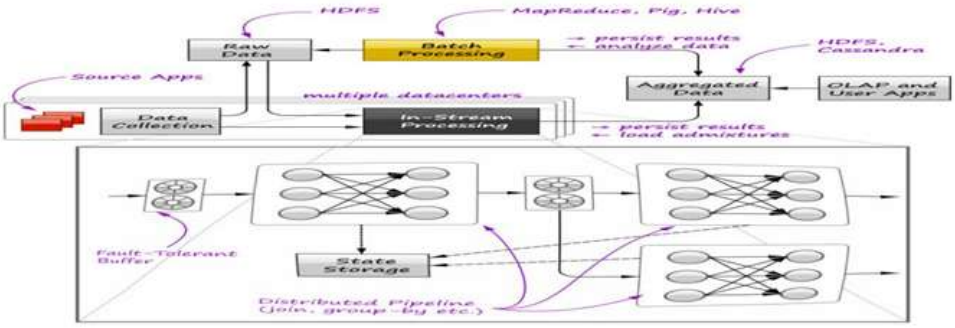

Fig.3. High level over view of Big Data Environment

- SQL-like functionality. The engine has to evaluate SQL-like queries continuously, including joins over time windows and different aggregation functions that implement quite complex custom business logic. The engine can also involve relatively static data (admixtures) loaded from the stores of Aggregated Data. Complex multi-pass data mining algorithms are beyond the immediate goals.

- Modularity and flexibility. It is not to say that one can simply issue a SQL-like query and the corresponding pipeline will be created and deployed automatically, but it should be relatively easy to assemble quite complex data processing chains by linking one block to another.

- Fault-tolerance. Strict fault-tolerance is a principal requirement for the engine. As it sketched in the bottom part of the figure, one possible design of the engine is to use distributed data processing pipelines that implement operations like joins and aggregations or chains of such operations, and connect these pipelines by means of fault-tolerant persistent buffers. These buffers also improve modularity of the system by enabling publish/subscribe communication style and easy addition/removal of the pipelines. The pipelines can be stateful and the engine's middleware should provide a persistent storage to enable state checkpointing. All these topics will be discussed in the later sections of the article.

- Interoperability with Hadoop. The engine should be able to ingest both streaming data and data from Hadoop i.e. serve as a custom query engine atop of HDFS.

- High performance and mobility. The system should deliver performance of tens of thousands messages per second even on clusters of minimal size. The engine should be compact and efficient, so one can deploy it in multiple datacenters on small clusters.First, we explore relations between in-stream data processing systems, massive batch processing systems, and relational query engines to understand how in-stream processing can leverage a huge number of techniques that were devised for other classes of systems.

- Second, we describe a number of patterns and techniques that are frequently used in building of in-stream processing frameworks and systems. In addition, we survey the current and emerging technologies and provide a few implementation tips.

4.1 Stream Replay

Ability to rewind data stream back in time and replay the data is very important for in-stream processing systems. This is the only way to guarantee correct data processing. Even if data processing pipeline is fault-tolerant, it is very problematic to guarantee that the deployed processing logic is defect-free. One can always face a necessity to fix and redeploy the system and replay the data on a new version of the pipeline. The system should work fast enough to rewind the data back in time, replay them, and then catch up with the constantly arriving data stream.

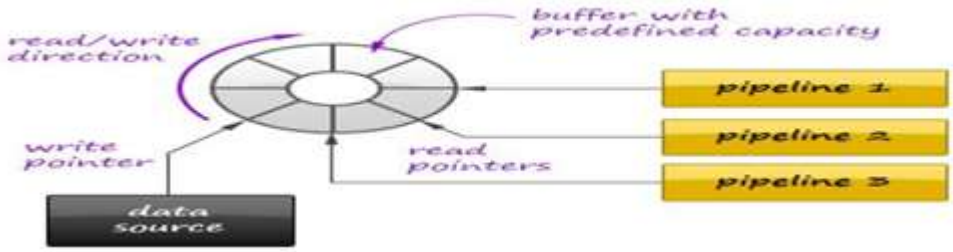

Fig 4. Big Data Stream Replay

4.2 Towards Unified Big Data Processing

It is great that the existing technologies like Hive, Storm, and Impala enable us to crunch Big Data using both batch processing for complex analytics and machine learning, and real-time query processing for online analytics, and in-stream processing for continuous querying. The key observation is that relational query processing, Map Reduce, and in-stream processing could be implemented using exactly the same concepts and techniques like shuffling and pipelining.

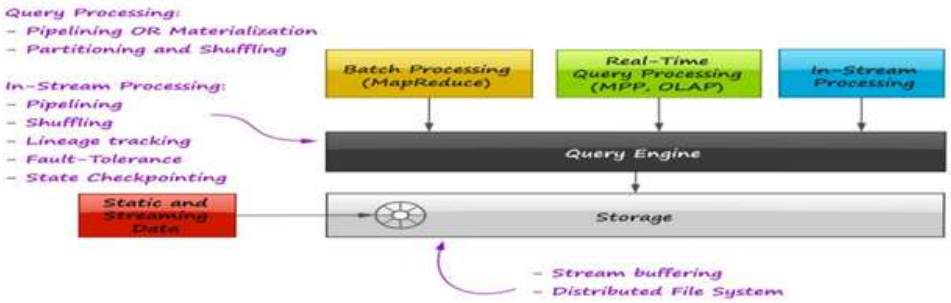

Fig. 5. Unified Big Data processing

5. A generic real-time analytic system

One major problem with streaming system is lack of flexibility. The architecture of *probability node*, *classification node* and *clustering node* is tailored for the Naive Bayes model. If we want to adopt a more comprehensive model to improve the accuracy, we need to completely change the design. Moreover, it is impossible to extend the system to support other analytic jobs such as game log analysis and social community detection. In summary, the architecture is limited to the spam detection system using Naive Bayes model. Another issue is the scalability and load imbalance problem. The interface is required to be compatible with Hadoop to reduce the efforts of migration, as most existing analytic jobs of Netease are processed in Hadoop with in – memory processing, processing updates and deployment of multiple data centers.

6. Conclusions

In this paper, we use the information security system in big data system evolves when users' requirements keep changing. How to design a generic system that can provide near real-time analytic services for many related applications, such as spam detection, game log analysis and social community mining. Based on our experiences, no solution can address all big data problems, especially when data size keeps increasing, more complex user requirements need to be handled, the emergence of new hard- ware violates the old design and the old system becomes too complicated for maintenance. New applications will emerge when we combine big data techniques with other conventional industries while in the combination process those applications will pose new requirements for big data systems, pushing us to search and propose new solutions.

Information security in big data environment is a promising fields in information security. This paper introduces impact to information security from two aspects of big data and cloud computing. In general, improving system efficiency and provide general cloud storage functions on premise to ensure user data and access authority are the Research direction of future safe cloud computing. At present, more things need to be done in cryptograph searching and reduplicate data removing.

After all, there is an urgent need of improved solutions concerning the users to control the use of their data and more research should be done in this field and there is also a need for more robust approaches in key management limitation, which could extend traditional approaches to Cloud computing.

References

[1]. A Navint Partners White Paper, "Why is BIG Data Important?" May 2012, http://www.navint.com/images/Big.Data.pdf.
[2]. http://www.informatioweek.com/software/business-intelligence/sas-gets.hip-to-hadoop-forbigdata/ 240009035? / pgno.2.
[3]. Chen Mingqi, Jiang He. USA Information Network Security New Strategy Analysis in Big Data [J]. Information Network Security. 2012(8):32—35
[4]. Narayanan A, Shmatikov V. How to break anonymity of the Netflix prize dataset. ArXiv Computer Science e-prints, 2006, arXiv:cs/0610105: 1-10
[5]. G. Caruana, Maozhen Li, Man Qi, A mapreduce based parallel svm for large scale spam filtering, in: Fuzzy Systems and Knowledge Discovery, FSKD, 2011, pp. 2659–2662. [6] Alfons Kemper, Thomas Neumann, Hyper: a hybrid oltp&olap main memory database system based on virtual memory snapshots, in: ICDE, 2011,pp. 195–206.
[6]. Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik-Lun Lee. Exploiting geographical influence for collaborative point-of-interest recommendation.//Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval(SIGIR'11), Beijing, China, 2011: 325-334

[7]. Goel S., Hofman J.M., Lahaie S., Pennock D.M. and Watts D.J.. Predicting consumer behavior with Web search. National Academy of Sciences, 2010, 7 (41): 17486– 17490
[8]. http://www.wired.com/science/discoveries/magazine/16-07/pb_theory Study Finds Web Sites Prying Less: Shift May Reflect Consumer Concerns[EB/OL]. http://www CNN.com, 2002-03-18
[9]. A survey of data disclosing in 2010 by Verizon[EB/OL].[2012-05-10]. Bessani A, Correia M, Quaresma B, et al. DEPSKY: Dependable and secure storage in a cloud-of clouds [C] //proc of the 6thConf on Computer System. New York: ACM, 2011:31-46
[10]. Sweeney L..k-anonymity: a model for protecting privacy. InternationalJournal on Uncertainty, Fuzziness and Knowledge-based Systems, 2002, 10 (5): 557-570 [12]Sweeney L..k-Anonymity: Achieving k-Anonymity Privacy Protection using Generalization and Suppression.
[11]. AshwinMachanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. L-diversity: Privacy beyond k-anonymity. ACM Transactions on Knowledge Discovery from Data, 2007, 1(1):1-52