Performance Evaluation of Unstructured PBRA for Bigdata with Cassandra and MongoDB in Cloud

Sangeeta Gupta, JNTU Kakinada, Hyderabad, India

ABSTRACT

In this article, performance evaluation of web collection data in data stores, such as NoSQL-Cassandra and MongoDB is presented, yielding scalability of applications. In addition to scalability, security of NoSQL databases remains highly unproved. It is noteworthy that existing works in the area of cloud with NoSQL focus on either scalability or security but not both aspects. Also, security, if provided, is at minor interface levels. In this article, the PBRA system is designed to deal with highly unstructured big data emerging from the twitter social networking service, which is new of its kind to strengthen the bigdata security. PBRA is Passphrase Based REST API model where the REST API methods are integrated with the user generated passphrase in addition to the private key for a set of records of user desirable number before storing into the Cassandra and MongoDB databases. Results are presented to illustrate the same for nearly 1 million records and the efficiency of Cassandra over MongoDB is observed. It is observed from the results that though the time taken to load and retrieve bulk data records is higher than dealing with cipher text, Cassandra performs better than MongoDB with the proposed security model.

KEYWORDS

Bigdata, Cassandra, Cloud, Decryption, Encryption, MongoDB, NoSQL, REST API

1. INTRODUCTION

Cloud computing allows end users to utilise the resources like hardware, software, servers etc. on a demand-driven basis, unlike grid and cluster computing which are the traditional approaches to access resources. Enormous amounts of data flooded across the internet and the storage capacities of the relational technologies have experienced inadequacy to access the huge amounts of data. To store petabytes (One quadrillion bytes) of data, most of the organisations, particularly social networking sites and e-commerce sites are moving towards the cloud to deploy their applications, but at increased security risks. Security is a major concern for IT enterprise infrastructures. It is critical to understand the importance of security as massive amounts of data termed as bigdata is processed and analyzed.

The foremost benefit of the cloud is to pay only for the resources which users utilise. If there is an unexpected set of users competing for access to resources, they would just have to pay only for what they have been using with every user’s request being satisfied. This is known as elasticity of the cloud. The cloud provides a variety of service models, such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), Database as a Service (DaaS) and other deployment models, such as public, private, hybrid and community clouds. To be hosted on a scalable environment, an application can use either of these models in a cost-efficient manner to realise benefits.
Other benefits provided by the cloud can be utilised in terms of elasticity, scalability, efficiency and reusability (Sandholm and Lee, 2014).

Big data is the growing amounts of data which are too big and complex to capture, store, process, and interpret. It is characterised by the 4V’s, such as Volume, Velocity, Veracity and Variety. The storage and analysis of such data can be made effective using the NoSQL databases (Gudiwada, Rao, and Raghavan, 2014).

Most modern world data is processed in the form of word documents, pdf files, audio and video formats. Relational databases may not be suitable to serve such data. Also, using relational databases for scalable applications impose heavy costs and make them less attractive for deploying large scale applications in a cloud (Agarwal, Das, and Abbadi, 2011). An alternate approach is to use the emerging NoSQL databases, which are not ACID-compliant. Atomicity requires actions (read/write) to be either fully complete or not done at all. Consistency ensures only valid data is to be stored in database. Isolation ensures that concurrent execution of actions results in a system state that would be obtained if actions are executed serially. Durability ensures that the committed actions will remain so in the event of system failures. In contrast to relational databases, NoSQL provides support to structured, unstructured, and semi-structured storage of massive data in terms of peta bytes. However, in NoSQL databases, current security trend is weak in nature, authentication and encryption is almost nonexistent. Authentication if exists, is not enabled by default in most of the NoSQL data stores. External encryption tools cannot be used, and they are vulnerable to SQL injection attacks. Based on the user requirements to provide security for chosen NoSQL data stores, this paper focuses on the various levels at which security can be provisioned by throwing a light on the security limitations that can motivate people to design solutions to overcome the limitations.

Amazon is the most preferable cloud service provider with a set of free tier usage options. Anyone with a credit/debit card login credentials can simply sign-up and access the services offered by Amazon. Bitnami integrated with Amazon is used to implement the work where Amazon EC2 (Elastic Compute Cloud) is the chosen platform to create Cassandra and MongoDB virtual machines onto which bulk data from twitter services is loaded (EC2, n.d.).

The methodology of this paper is organised as follows: The first section is the introduction. The second section presents related work or a background study, including the scenarios where the NoSQL databases Cassandra and MongoDB are used. Literature survey is carried out to identify the modern world scenarios of the applications using the aforementioned NoSQL databases. Majority of the works on these NoSQL databases exist in theoretical aspects rather than drawing attention towards practical orientation. This feature is highlighted in the current work in terms of developing an architectural model to deal with scalability and security aspects pertaining to twitter bigdata, as used by huge crowd. The third section presents the proposed architectural model designed with suitable modules to deal efficiently with voluminous data emerging across the web and also to perform comparison between Cassandra and MongoDB for scalable secure data. The fourth section presents the results as obtained when inserting and retrieving records in multiples of hundred expounding plain text and cipher text content in Cassandra and MongoDB. Performance changes are also brought into consideration before and after encrypting and decrypting the voluminous data and improvement in efficiency is observed for Cassandra over MongoDB. The fifth section concludes the work and throws light on the future directions.

2. RELATED WORK

Several works using NoSQL databases pertaining to scalability are discussed in this section and their limitations are observed. Many of the existing works in the literature present a comparative analysis between the relational and NoSQL databases, rather than discussing about the implementation and methodology suitable to solve a real-world streaming data maintenance and management problem.
Creating Effective Customer Solutions: A Process-Oriented Perspective
Ferdinand Burianek, Sebastian Bonnemeier and Ralf Reichwald (2013). Best Practices and New Perspectives in Service Science and Management (pp. 16-30).
www.igi-global.com/chapter/creating-effective-customer-solutions/74984?camid=4v1a

Using the Critical Incident Technique to Identify Factors of Service Quality in Online Higher Education
www.igi-global.com/article/using-critical-incident-technique-identify/47092?camid=4v1a