Initial Encryption of Large Searchable Data Sets using Hadoop

Krishna Sindhuja Kalusani

USC ID: T25568677

The amount of data generated each day is exponentially increasing and hence the data storage locations also. The data stored by the enterprise databases may be of terabytes and the security of the confidential data rather is a huge concern rather the whole legacy data has to be protected. This paper focuses on the encryption technique i.e. to encrypt all the data that is going into the database in the minimal time frame possible. Considering the time constraint and the velocity at which the data is being generated, the encryption technique proposed in the paper can be used in parallel using Hadoop Cluster. For this to implement, the encryption technique can be carried out in 5 steps:

- the Hadoop set up
- target preparation
- source data import
- encrypting the data
- exporting it to the target

Since the cloud storage can be accessible to everyone, it is essential to provide external solutions for the security of the sensitive data. Due to the large amount of data it may take long time to encrypt all the data in the databases. So the process should be transparent to the application and should happen parallel like in “CryptDB”. For step "import data to cluster", Apache Sqoop is used. For step "encryption", Map-Reduce is used. In step "export data to database", there are two potential methods. The first one uses Sqoop too, and the second one is a naïve method where the data is copied to the database with a database-specific import command.

Security of Sharded NoSQL Databases: A Comparative Analysis

Sharding is a key feature of databases which is a scaling process used for fast reads and writes in the databases. NoSQL is one of the databases that has ease to scale out because of their flexible schema and BASE properties. Yes, but is the distribution on various channels and servers secured? Is a concern and challenge that the database research community is facing. In this paper a detailed review of the security features offered by NoSQL databases is studied and proposes an assessment criterion. Hence facilitating the organisation to choose their databases wisely. Autosharding is the automatic and native horizontal distribution of data among different severs in NoSQL databases which, in turn, increase the performance and throughput of database operations. Auto sharding is one key concept that makes NoSQL standout among many other databases making it scalable. Other striking feature of auto sharding is load balancing. But security is lacking in NoSQL due to unauthorized exposure of backup and replicated data, insecure communication over the network and many more.

The assessment criteria that affect the security of the NoSQL databases are:

- Authentication
- Access Controls
- Secure Configurations
- Data Encryption
- Auditing

The security features have been analysed for the various open source, NoSQL sharded databases like MongoDB, CouchDB, etc.

Transforming Provenance using Redaction

Provenance records the history of a document for ensuring both, the quality and trustworthiness; while redaction identifies and removes sensitive information from a document. In the paper, the authors propose a graph grammar method approach for rewriting redaction policies over provenance. The rewriting procedure proposed converts a high level specification of a redaction policy into a graph grammar rule that transforms a provenance graph into a redacted provenance graph. This can be implemented using Semantic Web technologies. Quality of information along with correctness of the data determines the trustworthiness of the shared information. In lieu with the HIPAA rules for data security and disclosure, redaction as important as creating data. Redaction policy execution is a kind of access control mechanism, but choosing the graph method over traditional access control mechanisms is to identify those resources of the graph that a user is permitted or denied to view.

The currently available redaction tools delete the sensitive information of the documents such as images or text. These tools are not applicable to provenance since provenance is a directed acyclic graph (DAG) that contains information in the form of nodes and relationships between nodes. In this paper, the authors apply a graph transformation technique (generally called graph grammar which is flexible enough to perform fine-grained redaction over data items and their associated provenance graphs. A prototype is implemented that performs redaction over the information resources in the graph. This prototype uses an interface which mediates between our graph transformation rules and a high-level user policy specification language. This interface allows us to separate the business rules from a specific software implementation, thus promoting easier maintenance and reusability.