DALHOUSIE UNIVERSITY

CSCI 5408 – Data Management, Warehousing, Analytics

Assignment – 5

Prashit Patel - B00896717

Security and Privacy Solutions associated with NoSQL Data Stores

Summary

The given research paper focuses on the security and privacy solutions for NoSQL databases. It provides reasons behind the need for NoSQL databases. It further compares NoSQL databases with relational databases based on different factors. It then proposes problems related to privacy and security in NoSQL databases and offers a solution that can solve those issues.

The need for a NoSQL database or "Not Only SQL" database arose with the advances in cloud computing and distributed applications. They support large volumes of data storage and processing, which is currently the need of the hour as lots of data is generated with each passing day. NoSQL databases provide high availability, scalability, and better performance for storing large-scale data. It supports both semi-structured and non-structured data and is faster than traditional databases.

But the disadvantages of NoSQL databases are critical too. They lack encryption support and has poor authentication. Also, it does not support role-based access control (RBAC) for simple authorization, making it vulnerable to injections and denial of service attacks. Also, according to the CAP theorem of Brewer, a distributed system can only provide at most two from consistency, availability, and partition tolerance. Thus, all the advantages of NoSQL databases cannot be justified simultaneously. The critical issues for NoSQL databases presented in this paper are data protection and access control.

Relational databases use tables that consist of rows and columns to store structural data. Whereas, the NoSQL database does not have any systematic approach and can be store any data like non-structured or semi-structured. New fields can also be added during runtime. Comparison can be made based on different factors as below:

- Reliable Transactions: Relational databases support ACID (Atomicity, Consistency, Isolation, Durability) for transactions, whereas NoSQL databases do not.
- Scalability Issues and Cloud Support: NoSQL databases are fully supported by cloud environments. Scalability is a significant issue for relational databases as they scale vertically while NoSQL databases scale horizontally, which can be easier and faster.
- Complexity and Big Data Management: Relational databases are complex as database structure has to be created, which is a complicated task. Considering big data management, NoSQL databases provide higher speed for data management across distributed nodes and easily support large amounts of data.
- Data Model: Relational databases have well-defined structures for tables. NoSQL databases can use different models such as graphs, key-value pairs, and documents. It is schema-less and thus can store non-structured or semi-structured data.

- Data Warehouse and Crash Recovery: NoSQL databases only can handle large data as the focus is on scalability, availability, and high performance rather than structure. Crash recovery is supported by recovery manager and log file in relational databases, whereas in NoSQL databases, it depends on replication for recovery.
- Privacy and Security: Relational databases do not provide any security features embedded in the database. For NoSQL databases, this is the most critical shortcoming as data files are not encrypted by default. Also, distributed systems increase the attack surface, and enforcing integrity rules is more complex.

The paper provides two solutions to privacy and security issues for NoSQL databases.

1. Pseudonyms based Communication Network:

   Users need to log in once only and can only be identified by their pseudonyms. Transactions carried out cannot be linked to the same user as identity is disclosed. Identity Providers (IP) provide identity to the user and digitally sign the credentials using its own public and secret key. IP generates unique random keys and assigns pseudonyms to users.

   Credential Authority (CA) ensures users have public and private keys. The user sends its public key and digital identity of CA for verification where the digital identity of CA shows they know the secret key. Verifier verifies the user's identity and communicates the validity status with CA. Service Providers (SP) have their independent public and private keys linked to users via pseudonyms.

   This system protects user privacy as Service Providers only have pseudonyms of users. IP adds a random number with each user request which SP decodes each time to ensure unlikability of the pseudonym. If the user abuses the system, SP blacklists and reveals those random numbers so that IP can revoke the user's pseudonym. This way, it ensures complete privacy and security for the user.

2. Monitoring, filtering, and blocking:

   In a cloud environment, no information is logged for communication of clusters, user information or data alteration. Along with providing security, this also prevents identifying data breaches or malicious data loss in a cluster. Real-time security mechanisms exist to perform real-time analysis to detect anomalies but are limited to some APIs only. But in general, initial authentication can be achieved via Kerberos to ensure filtering and blocking.

   This report perfectly describes NoSQL systems, their advantages, privacy, and security concern and provides a detailed solution for ensuring privacy and security for NoSQL databases. Thus, technical concepts are clearly explained and do not need further improvements.

References:

1. G. Vonitsanos, E. Dritsas, A. Kanavos, P. Mylonas and S. Sioutas, "Security and Privacy Solutions associated with NoSQL Data Stores," *2020 15th International Workshop on Semantic and Social Media Adaptation and Personalization (SMA)*, 2020, pp. 1-5, doi: 10.1109/SMAP49528.2020.9248442.
   Paper Link https://ieeexplore-ieee-org.ezproxy.library.dal.ca/document/9248442