ISSN 2347 - 3983

Volume 9. No. 4, April 2021

International Journal of Emerging Trends in Engineering Research

Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter13942021.pdf

https://doi.org/10.30534/ijeter/2021/13942021

Data integrity issues and challenges in next generation non-relational document-oriented database outsourced in public cloud

Mujeeb-ur-Rehman Jamali[1], Abdul Ghafoor Memon[2], Nadeem A. Kanasro[3], Mujeeb-u-Rehman Maree[4]
IMCS, University of Sindh, Jamshoro, Pakistan mujeebjamali@usindh.edu.pk[1]
IMCS, University of Sindh, Jamshoro, Pakistan ghafoor@usindh.edu.pk[2]
SU CL &IMCS, University of Sindh, Jamshoro, Pakistan nadeem.kanasro@usindh.edu.pk1[3]
IMCS, University of Sindh, Jamshoro, Pakistan mujeeb@usindh.edu.pk[4]

ABSTRACT

Database as a Service provides high availability and scalability in cloud but users do not physically control over their data, therefore, data integrity is serious concerned. Authenticity of data and its verification is one of major risk in next generation document-oriented databases. It is possible that malicious insider and outsider can change and compromise the data. A system is proposed in which cloud environment secure storage and access semi-structured data for non-relational document-oriented database. The proposed system effectively provides data integrity for sensitive and confidential fields and verification of data whether it has been altered or not in outsource database in the public domain.

Key words: Cloud, document database, data integrity, security.

1. INTRODUCTION

Cloud is new paradigm where Internet use for on-demand resources access ad provision rapidly. Using the services of cloud data can be access from anywhere and anytime 24/7. In the cloud renting services (S) are provided i.e., infrastructure, platform, database and software, which is known as XaaS [1].

NoSQL databases also known next generation database which provides facility to provides services over cloud. Database as a Service is cloud service. These databases not support SQL for manipulation of data and performing queries. These are high performance in read and write operation, very flexible, schema-less, simple data model and highly scalable. These databases can process very large amount of data efficiently compare to traditional relational database. There is no transaction and does not follow the ACID properties. BASE is followed where all updates are eventually stored after some time in all replicas of databases [2].

Types of non-relational databases included graph, column, document and key-value. These databases store data in the format of structured, unstructured and semi-structured. The key-value is fastest database where each attributes of has corresponding value and types e.g., Riak, Amazon S3 and Dynamo [3]. In column database as traditional database where row have different number of column with super column e.g., Cassandra and HBase. In Graph oriented stores data as node and edge. It has nonlinear structure e.g., Ne04J and InfoGrid.

DOCUMENT-ORIENTED DATABASES

Document database use data model of BSON which is similar to JSON format for management of semi-structured data. These databases not required declare schema before insertion data and not fixed schema and it also not required predefined data type of data [4]. It is flexibility and strength of document data where collections do not have fixed structure with unique attributed can be used. In the collection document may be varies structure and it may be complex structure with nested document, array and embedded document. Document databases have simple data model which is scalable and support high volume of data storage and provide high performance for CRUD operations [5]. Distributed scale out architecture is used where commodity hardware / software utilized instead of scale-up system to increase the system resources.

In document-oriented database data at rest is not secure, data are kept unencrypted by default, therefore data integrity is serious concerned and authenticity of data is risk. The malicious cloud service provider staff or outsider unauthorized can modify data. These databases are vulnerable and not proper data protection mechanism.

2. BACKGROUND

In this section research papers are presented that propose data integrity. In this research [6] presented that alteration of data is major concern in data reside on cloud. In this paper [7] presented that data encryption is not supported by most NoSQL document databases thus developers to implement cryptographic methods at application level or middleware layer for protection of data integrity. A techniques was proposed [8] for protection, correctness, completeness and verification included hash function, secret keys for data encryption and data authenticity and verification of data authentication Hash message authentication codes (HMAC) used. Data Encryption is done using symmetric encryption i.e, AES. HMAC value along with signed data stores in the cloud for data integrity in outsourced databases. In this research [9] presented that data encryption of document model database not achieved due to not support of encryption. Authors [10] presented that data integrity is not always achieved in non-relational database due to eventually consistent one of the BASE properties. It was argued that data is stored as clear text and readable format thus, data confidentiality is not achieved and user's authentication can be achieved using some of external method. In this research [11] proposed that RSA and Elliptic Curve are rigorously used public key cryptographic algorithms, it is further proposed that ECDSA is better performance than RSA because it provides the same level of security with less key size. In this research, the proposed work is difference using enhance modern cryptography i.e., asymmetric public key cryptography for provision of data integrity.

3. DATA INTEGRITY TECHNOLOGIES

The content of data is checked and verified using digest for data integrity. If data is altered, then digest of data will not verify. Fixed size / condense form of data is Message Digest. Hashing is used to convert data of various size into fixed size hexadecimal number depend of the algorithms that represent the original data. It produces random hexadecimal number Hash functions are irreversible and it is possible brute-force attack try every possible input. Hash function not produce the same hash value from two different inputs if it produces same it is a collision and weakness of hash algorithm. There are various hash functions used in cryptography including Message Digests and Secure Hash Algorithms. MD algorithm was developed by Ron Rivest. MD2 produces 128-bit (16-byte) digest. Due to collisions and weaknesses in MD2, MD3 and MD4 a new and enhance version is MD5, it is also 16-bytes digest algorithm which is faster and stronger and no any collisions found. SHA-1 is improved and stronger version SHA. It produces longer digest 128, 160, 192, 256, 384 & 512 bits and increase in size of digests make it safe from cryptanalysis attacks.

To verify and prevent the alteration of content digital signature is used. The most prominent digital signature algorithms are RSA, DSA and ECDSA. Public-key cryptography superior in security to provide authenticity of data and identity.

The security strength of DSA is based on discrete logarithm in finite field, it consists three algorithms use for key pair generation (Private and Public Key), signing algorithm and a signature verifying algorithm. RSA algorithm is based on Integer Factorization Program. RSA private key is used for signing the encrypted data, meanwhile public key is used for verification the same for the purpose of data integrity. The security hardness of ECDSA based on Elliptic Curve discrete logarithm problem in finite field using the various curve specifications using the smaller key size with equal level of security as compare to RSA and DSA. These all are computationally infeasible no any successful attack found in literature.

4. METHODOLOGY

In this paper, public key cryptography well-known algorithms i.e., DSA, RSA and ECDSA used to provide integrity of data using various message digest and secure hash algorithms with different size. RSA and ECC algorithms encrypt data into unreadable format and digital signature is attached with data to be verified the alternation of data and only proper private key can decrypt data into plain text and public key is used to verify the change and alternation.

In the proposed work**(Figure 1)**, secure middleware carried out the required transformation and forward the digital signature alongwithciphertext to document database. On the receiving ciphertextalongwith digital signature secure Middleware verifies integrity of data has not been altered, if the data was altered, the check will also be different. It is ensured that the check value cannot be altered to match any changes in the data. If the check value shows no alterations, the data has been shown to have integrity.

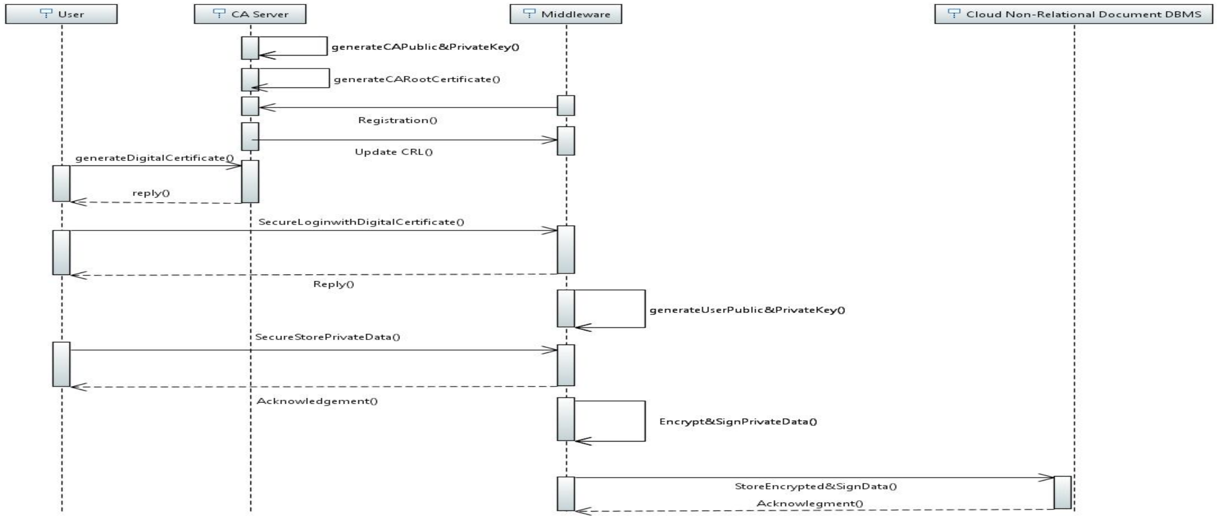

User
CA Server
Middleware
Cloud Non-Relational Document DBMS
generateCAPublic&PrivateKey()
generateCARootCertificate()
Registration()
Update CRL()
generateDigitalCertificate()
reply()
SecureLoginwithDigitalCertificate()
Reply()
generateUserPublic&PrivateKey()
SecureStorePrivateData()
Acknowledgement()
Encrypt&SignPrivateData()
StoreEncrypted&SignData()
Acknowlegment()

Figure 1: Sequence Diagram of Proposed System

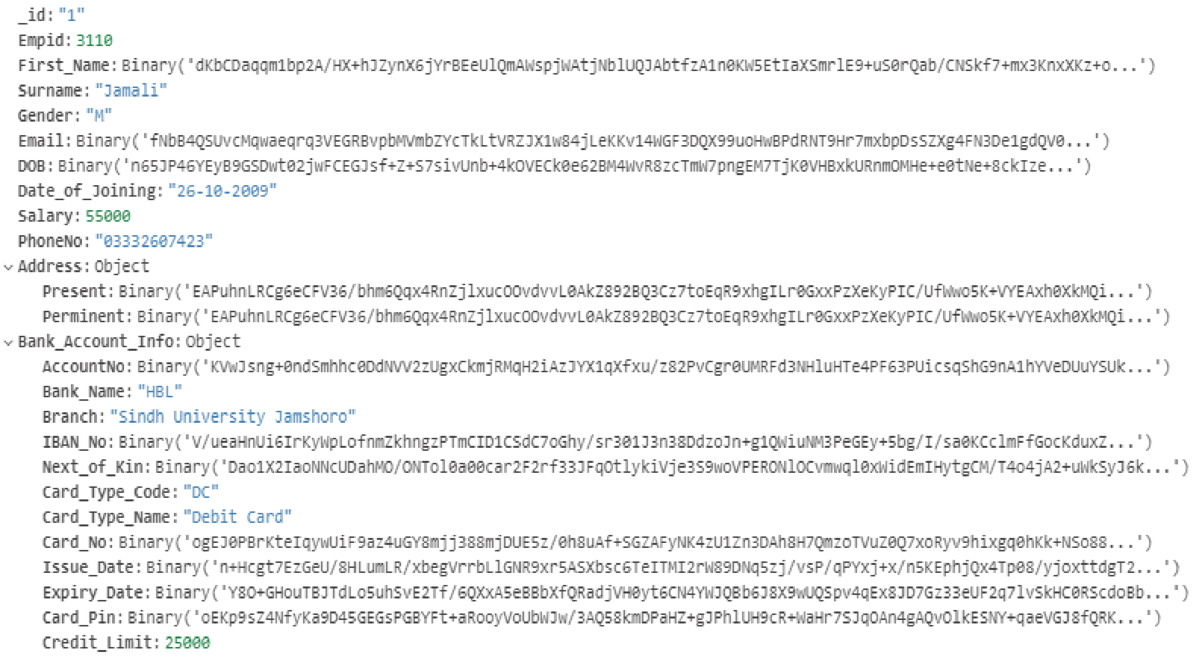
- Middleware encrypts the sensitive fields using Public Key (CF = EncPKU (PF)) PKu is public key.
- Middleware signs the encrypted fields with Private Key SK (SigF=SigSK(CF))
- Middleware stores encrypted and signed private and sensitive fields (OE=EncPKU (CF, SigF)) to document-oriented database in public cloud.
- User makes access request to Middleware and retrieve the encrypted data alongwith digital signature from document-oriented database.
- Middleware performs decryption operation thereafter verification of signature of SigF that prove encrypted data is valid or not.

5. HARDWARE AND SOFTWARE REQUIREMENT

System prototype implemented and experiments were conducted using Hardware i.e., M-5 Y 10c (4 CPUs) 0.8 GHz Processor, Primary Memory 2 GB, HDD 500 GB and software's i.e., Window 10 (64-bit). MongoDB 3.4, JCE and JCA API of Java language for asymmetric cryptographic.

6. RESULTS AND DISCUSSION

Projection of the document in the collection of the database with complex document fields, array and embedded documents. In **(Figure 2)** shows sensitive fields of the document in encrypted and unreadable form and non-sensitive data as plaintext in the collection of document database. Transformation of data taken place by secure Middleware using the application level encryption in cloud environment outsourced in public domain [12, 13].



```
_id: "1"
Empid: 3110
First_Name: Binary('dKbCDaqqm1bp2A/HX+hJZynX6jYrBEeUlQmAWspjWAtjNblUQJAbtfzA1n0KW5EtIaXSmrlE9+uS0rQab/CNSkf7+mx3KnxXKz+o...')
Surname: "Jamali"
Gender: "M"
Email: Binary('fNbB4QSUvcMqwaeqrq3VEGRBvpbMVmbZYcTkLtVRZJX1w84jLeKKv14WGF3DQX99uoHwBPdRNT9Hr7mxbpDsSZXg4FN3De1gdQV0...')
DOB: Binary('n65JP46YEyB9GSDwt02jwFCEGJsf+Z+S7sivUnb+4kOVECk0e62BM4WvR8zcTmW7pngEM7TjK0VHBxkURnmOMHe+e0tNe+8ckIze...')
Date_of_Joining: "26-10-2009"
Salary: 55000
PhoneNo: "03332607423"
⌄ Address: Object
    Present: Binary('EAPuhnLRCg6eCFV36/bhm6Qqx4RnZjlxucOOvdvvL0AkZ892BQ3Cz7toEqR9xhgILr0GxxPzXeKyPIC/UfWwo5K+VYEAxh0XkMQi...')
    Perminent: Binary('EAPuhnLRCg6eCFV36/bhm6Qqx4RnZjlxucOOvdvvL0AkZ892BQ3Cz7toEqR9xhgILr0GxxPzXeKyPIC/UfWwo5K+VYEAxh0XkMQi...')
⌄ Bank_Account_Info: Object
    AccountNo: Binary('KVwJsng+0ndSmhhc0DdNVV2zUgxCkmjRMqH2iAzJYX1qXfxu/z82PvCgr0UMRFd3NHluHTe4PF63PUicsqShG9nA1hYVeDUuYSUk...')
    Bank_Name: "HBL"
    Branch: "Sindh University Jamshoro"
    IBAN_No: Binary('V/ueaHnUi6IrKyWpLofnmZkhngzPTmCID1CSdC7oGhy/sr301J3n38DdzoJn+g1QWiuNM3PeGEy+5bg/I/sa0KCclmFfGocKduxZ...')
    Next_of_Kin: Binary('Dao1X2IaoNNcUDahMO/ONTol0a00car2F2rf33JFqOtlykiVje3S9woVPERONlOCvmwql0xWidEmIHytgCM/T4o4jA2+uWkSyJ6k...')
    Card_Type_Code: "DC"
    Card_Type_Name: "Debit Card"
    Card_No: Binary('ogEJ0PBrKteIqywUiF9az4uGY8mjj388mjDUE5z/0h8uAf+SGZAFyNK4zU1Zn3DAh8H7QmzoTVuZ0Q7xoRyv9hixgq0hKk+NSo88...')
    Issue_Date: Binary('n+Hcgt7EzGeU/8HLumLR/xbegVrrbLlGNR9xr5ASXbsc6TeITMI2rW89DNq5zj/vsP/qPYxj+x/n5KEphjQx4Tp08/yjoxttdgT2...')
    Expiry_Date: Binary('Y8O+GHouTBJTdLo5uhSvE2Tf/6QXxA5eBBbXfQRadjVH0yt6CN4YWJQBb6J8X9wUQSpv4qEx8JD7Gz33eUF2q7lvSkHC0RScdoBb...')
    Card_Pin: Binary('oEKp9sZ4NfyKa9D45GEGsPGBYFt+aRooyVoUbWJw/3AQ58kmDPaHZ+gJPhlUH9cR+WaHr7SJqOAn4gAQvOlkESNY+qaeVGJ8fQRK...')
    Credit_Limit: 25000
```

Figure 2: Projection of Document Database (Compass View)

Hash Value, Message Digest and Digital Signature generated using the asymmetric algorithm thereafter inserted in non-relational document database. For obtaining proper data each algorithm run alteast 10 time. Mean time and standard deviation of each operation are presented in **Table 1** of the recorded frequency of each recorded results in millisecond. It was found that insertion of MD5 took 54.4 with dispersion of value from mean was 2.757, SHA1 took 54.9 with 3.247 dispersion, SHA-256 took 53.2 with 3.938, SHA-384 took 52.9 with 4.725 and SHA-512 took 50.7 with 1.252. In **(Figure 3)**it was revealed that SHA-512 bits took less time in millisecond i.e., 50.7 which less than message digest (128 bits) and secure hash algorithms (160, 256 and 384 bits).

Table 1: Insertion Message Digest, Hash Value and Digital Signature in Document Database

	MD5	SHA1	SHA-256	SHA-384	SHA-512
Mean Time (ms)	54.4	54.9	53.2	52.9	50.7
Standard Deviation	2.757	3.247	3.938	4.725	1.252

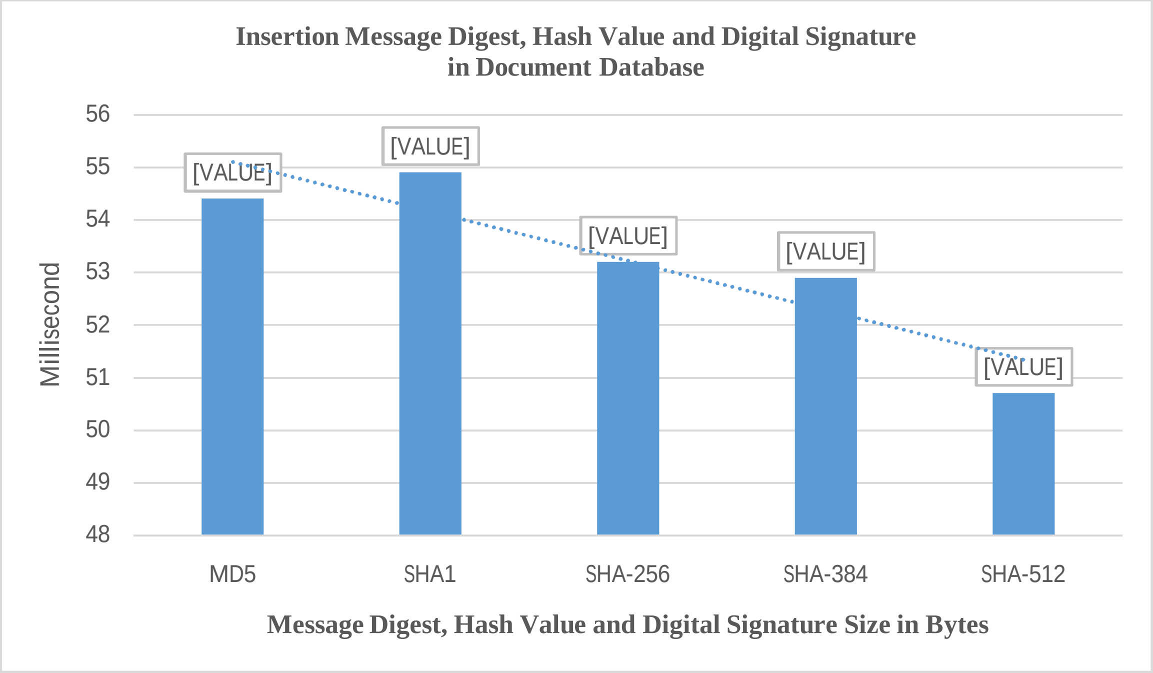


Figure 3: Insertion Message Digest, Hash Value and Digital Signature in Document Database

7. CONCLUSION

Cloud providers (malicious insider and outsider) can change users' data due to cloud users do not physically possess and control on their data, therefore, data integrity may be at risk. A secure Middleware is proposed for sensitive, confidential, private and personal identifiable information are being stored and the same to be checked and verified that the data has not been tampered stored in document-oriented database in public cloud. The proposed system provides enhance security of data using modern asymmetric cryptography according to user requirement.

8. ACKNOWLEDGEMENT

I would like to express my profound humble and gracious thanks to Prof. Dr. Abdul GhafoorMemon and Prof. Dr. Mujeeb-u-Rehman Maree who help and guide make it possible to complete this work. Thanks to all for supporting me throughout this work.

REFERENCES

[1] Mauro Femminella, Matteo Pergolesi and Gianluca Reali1, (2018). **"IoT, big data, and cloud computing value chain: pricing issues and solutions"**, Springer Annals of Telecommunications 73:511–520. Https://doi.org/10.1007/s12243-018-0643-6.

[2] DejunTeng, Jun Kong and Fusheng Wang, (2018). **"Scalable and flexible management of medical image big data"**, Springer Distrib Parallel Databases (2019) 37:235–250. Https://doi.org/10.1007/s10619-018-7230-8.

[3] Muhammad Younas, (2019). **"Research challenges of big data"**, Springer Service Oriented Computing and Applications (2019) 13:105–107. https://doi.org/10.1007/s11761-019-00265-x.

[4] InshaMearaj, PiyushMaheshwari and ManinderJeet Kaur, (2018), **"Data Conversion from Traditional Relational Database to MongoDB using XAMPP and NoSQL"**, IEEE Fifth HCT Information Technology Trends (ITT 2018), Dubai.

[5] Shady Hamouda and ZurinahniZainol, (2017), **"Document-Oriented Data Schema for Relational Database Migration to NoSQL"**, IEEE International Conference on Big Data Innovations and Applications. DOI 10.1109/Innovate-Data.2017.13.

[6] Ge Wu, Yi Mu, Willy Susilo, FuchunGuo and Futai Zhang, (2019). **"Threshold privacy-preserving cloud auditing with multiple uploaders"**. Springer International Journal of Information Security (2019) 18:321–331. Https://doi.org/10.1007 /s10207-018-0420-6.

[7] Ming-Hung Shih and J. Morris Chang, (2017). **"Design and Analysis of High Performance Crypt-NoSQL"**, EEE 978-1-5090-5569-2/17 (p-52-59).

[8] Grisha Weintraub and Ehud Gudes, (2017). **"Crowdsourced Data Integrity Verification for Key-Value Stores in the Cloud"**, IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing. DOI 10.1109/CCGRID. 2017.17.

[9] EbrahimSahafizadeh and Mohammad Ali Nematbakhsh, (2015). **“A Survey on Security Issues in Big Data and NoSQL”**. Advances in Computer Science an International Journal.

[10] Obay G. Altrafi, Mohamed A. Mohamed and Mohammed O. Ismail, (2014). **“Relational vs. NoSQL Databases: A Survey”**. International Journal of Computer and Information Technology.

[11] DhanashreeToradmalle, Rohan Singh, Het Shastri, Nikita Naik and Vishal Panchidi, (2018). **“Prominence of ECDSA Over RSA Digital Signature Algorithm”**, Proceedings of the Second International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)

(I-SMAC 2018) IEEE Xplore Part Number: CFP18OZV-ART; ISBN:978-1-5386-1442-6.

[12] M. R. Jamal, A. G. Memon, M. R. Maree, (2020). **“Security issues in data at rest in a non-relational Document** Database**”**, Sindh University Research Journal (Science Series), Vol. 52 (03) p-279-284. Http://doi.org/10.26692/sujo/2020.09.41.

[13] Mujeeb-u-Rehman, Xiaohu Yang, Jinxiang Dong, Memon Abdul Ghafoor, **“Heterogeneous and homogenous pairs in pair programming: An empirical analysis”,** Proceedings of Canadian Conference on Electrical and Computer Engineering (CCECE/CCGEI) Saskatchewan Canada, May (1-4) 2005, pp1116-1119. © 2005 IEEE.