Experimental evaluation of data security techniques used for big data

Dr. Mani Arora
Assistant Professor
Khalsa College,Amritsar
Punjab

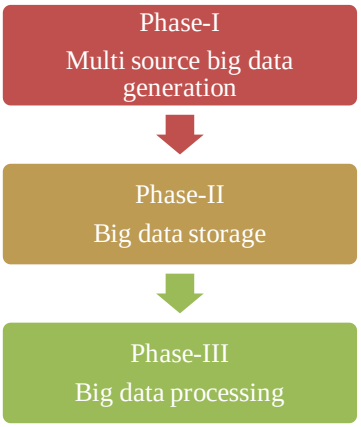
Abstract

At global level, the degree of connectivity of communication networks has been augmented due to which there has been an outburst in the growth of big data. Growth in data generated is exponential. The affluence of big data makes it alluring to cyber criminals, who are using ever more refined techniques to infringe organisations and by any means operate this information to steal it, often for deceitful use. This paper presents traditional as well as recent cryptographic techniques used for securing big data. Due to the hefty volume of big data being transferred, it is crucial that an encryption solution does not crash on the speed or performance of the network.

Keywords: Big data, Homomorphic encryption, Attribute based encryption, Cryptography, RSA, 3DES, Symmetric, Asymmetric, EMDS

Introduction

In the present system of data-centric world, big data dispensation and analytics have become significant to the majority enterprises and government applications. Big data term [1,2] exclusively means that data sets are formed from various sources in multiple formats with incredibly high speed. Numerous business houses use immense data as it has broad prospect across the globe, in the field of marketing and technical research devoid of having a glance for the vision of its security. Due to diverse security issues, there is infringe in the security of information. A set of codes called as SQL injection are conceded by the attackers and hackers to split the access of database. In order to certify big data privacy diverse techniques have been developed in recent years. There are generally three phases in big data life cycle [3]:



At some point in these three phases there are diverse types of security concerns. In **Phase-I** for the safeguarding of privacy, access restraints as well as techniques that can fabricate data are used. In **Phase-II** encryption based techniques are used. The bigger storage of data leads to ensure the biggest security of the data. The data processing **Phase-III** includes privacy safeguarding data publishing and knowledge withdrawal from the data. In privacy maintaining data publishing anonymization techniques are used such as simplification

furthermore suppression it makes adequate efforts to shield the privacy of data. In this paper we primarily focus attention on a depiction of significant security threats, notable breaches and an analysis on cryptographic tools. Finally, at the end of the paper we wrap up and have proposed a technique which can give better results in securing hefty amount of data in addition to transmission speed as big data relies heavily on high speed and secured data networks to transmit it.

Popular trends and Emerging threats

1. The swift expansion in IOT devices, which will create an unbroken flow of big data be at the rim of the network, numerous organizations do not deem on for them when assessing their cyber security. However if left unprotected, these devices are providing hackers with 20 billion opportunities to gain access to networks.
2. Cloud computing plays an important role in the collection storage and evaluating of big data, by their towering performance data networks that are at risk if offensively protected.
3. The forthcoming technology of quantum computing also plays a significant part in cyber security. Despite the fact that the enormous computing power of quantum computers will have a transformative outcome on computing, including big data analysis. Quantum computers will be able to smash current unrestricted key encryption algorithms in a fraction of the time taken by the traditional computing methods.

Notable Breaches

As escalating amounts of data surges across networks, it leaves it vulnerable to breaches ranging from hack attacks to internal data misconfiguration or loss. Nearly 31 million records were exposed in the 13 biggest breaches in the first half of 2019 with 11 of the top 13 breaches occurring at medical or health organization.

- Columbia surgical specialists learned on Jan9, 2019 that hackers had carried out a ransom ware attack aligned with the electronic systems of the Spokane, wash based healthcare facility. Numbers of records exposed were 400,000.
- In Nov 2018, Marriot hotels proclaimed a breach in which 500 million records were stolen. This enclosed the personal information of all Starwood hotels customers dating back to 2014, in some cases including credit card details and passport information.
- Between Aug1, 2018 and March 30, 2019 data is breached of hundreds of thousands of Bio Reference Laboratories customers which was stored on the web payment page of the American Medical Collection Agency (AMCA). AMCA is a peripheral collection Agency used by Lab Corp and other healthcare companies. Statistics information of results exposed was 422,600.

End to End Encryption

Encryption is an essential element in certifying the security of big data networks. The data need to be fully protected otherwise it may lead to insecure interface, sharing of resources, data leakage and inside attacks. It is required that cryptography techniques should be deployed as an end to end solution across all the layers of the network. In the occurrence of breach, encrypted data is incomprehensible by hackers and is therefore render useless. In addition, the presumptuous secrecy provided by encryption solutions prevents rogue data being inputted hooked on to systems.

Basic cryptographic methods

These consist of asymmetric and symmetric key encryption techniques.

- **Asymmetric Cryptography**

Public Key Cryptography takes into account two pairs of keys one for encryption and other for decryption. The key used for encryption is a public key and circulated as well as doled out. On the other hand, the key used for decryption is a private as well as exclusive key. Some most commonly used basic techniques are RSA, Diffie-Hellman(DH) key exchange, Elliptic curve cryptography, Hash functions like SHA-2,SHA-3family.

RSA (**Rivest–Shamir–Adleman**)

It is the first and foremost public-key cryptosystems and is extensively used for securing data transmission. In such a cryptosystem, this asymmetry is based on the practical difficulty of factoring the product of two large prime numbers, the "factoring problem". RSA is a comparatively slow algorithm, in addition to this; it is not as much commonly used to directly encrypt user data. More often, RSA passes encrypted shared keys for symmetric key cryptography which in turn can execute bulk encryption-decryption operations at a good deal higher speed. This technique use key size varying from 1,024 to 4,096 bits and can encrypt data as large as its key. It is quite slow technique and is not profcint for encrypting bulk data.

Diffie-Hellman(DH) key exchange

Diffie-Hellman key exchange, also called exponential key exchange, is a process of digital encryption that uses numbers lifted up to definite powers to generate decryption keys on the basis of components that are never directly transmitted. It is used for firmly exchanging cryptographic keys over a insecure communication channel.

Elliptic curve cryptography

It is an approach based on the algebraic structure of elliptic curves over finite fields. ECC requires smaller keys compared to symmetric and asymmetric encryption (based on

plain Galois fields) to provide equivalent security [4]. The most important advantage of using Elliptic Curve based cryptography is condensed key size along with it consequently speeds. Elliptic curve based algorithms makes use of significantly less significant key sizes as compared to other symmetric and asymmetric techniques.

Hash Functions

These are the techniques of creating message digests that are used for examining digital integrity required in applications akin to digital signatures in digital documents. The most generally known secure hash functions are SHA2 and SHA3.

- **Symmetric Key Cryptography**

The Symmetric Key Cryptography technique uses the identical key for encoding and decoding information for making it secretive. The sender as well as recipient of data ought to correspond in the same manner and share identical key and maintain information confidential preventing data access commencing extraneous factors. Some most commonly used techniques are 3DES, Blowfish and AES. EMDS[15] can also exhibit to be moderately advantageous technique in case of securing large amount of data as it compresses the size of cipher text.

3DES

Triple DES is used during Federal organizations in the course of sentry sensitive data. Security of data is done complete through transmission or at the same time as in storage may be requisite to sustain the discretion as well as consistency of the information representing through the data. The algorithm noticeably categorizes the mathematical steps indispensable to transform data hooked scheduled on a cryptographic cipher likewise to transform the cipher reverse to unique shape. By this technique plain text of 64 bits get converted to 64 bits cipher text. Key size 112 bits is used in this technique.

Blowfish

This encryption algorithm was devised noticeably by Bruce Schneier. Encryption algorithm gets hold of a plaintext of 64-bits block along with an incoherent length key as input and produces a cipher text of 64-bits block as output. It is suitable for applications like automatic file encryption where the key fails to alter frequently. The key size can contrast from 32 to 448 bits. This algorithm is one of the secure traditional encryption algorithms because both the sub keys as well as S boxes are created by algorithm itself.

Advanced Encryption Standard (AES)

The additional popular and broadly adopted symmetric encryption algorithm likely to be come across currently is the AES. It is initiated at least six time faster than triple DES. AES is an iterative technique rather than Feistel cipher [5]. It is hinged on substitution permutation network. It encompasses a series of concurrent operations, a few of which demand replacing inputs by specific outputs as substitution operation whereas others engage shuffle bits around as permutation operation. AES executes all its computations on bytes. Hence, AES treats the plaintext block as 16 bytes. These 16 bytes are arranged in four columns as well as four rows for dealing out as a matrix. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a unlike 128-bit round key, which is calculated from the original AES key[6].

EMDS Algorithm

EMDS [15] is a dictionary technique for encryption. It focuses both on security as well as size of transmitted data. Its intention is to reduce the size of cipher text next to security restrictions so that bandwidth consumption can be abbreviated. Furthermore, this will consume less memory space at destination machine. This algorithm for encryption and decryption uses two keys of dissimilar sizes to conserve a firm security. Though this algorithm uses dictionaries for encryption and decryption it can be a good candidate technique for big data security.

Advanced Cryptographic methods

The advanced asymmetric encryption techniques have diverse features in managing big data such as homomorphism encryption, identity based encryption, verifiable computation, attribute based encryption. Apart from these there is Format Preserving Encryption based on symmetric key encryption. A similar technique is Format Preserving Hashing performing secure hashing despite the fact of preserving the format of the plaintext.

Format Preserving Encryption

It is exceedingly fast technique as it performs operations on block ciphers. In this encryption technique during encryption format of data is not altered. It employs basic symmetric encryption techniques including AES. It is relatively appropriate for encrypting big data as well as grants confidentiality however its analytics capabilities are limited.

Format Preserving Hashing

It is the latest method to anonymize sensitive data. It endows with a flexible substitution between protecting the secrecy of data subjects. With this method an attacker would have to

reverse a hash function in order to recuperate an original value which is near to impossible to do. There is merely no furtive information, either in the form of a cryptographic key or authentication credentials that will let an attacker reverse format preserving hashing. So, from a definite point of view, the security properties of format preserving hashing are essentially better than encryption.

Homomorphic Encryption

In this technique algebraic operations like multiplication and addition are applied on the plaintext. This encryption technique has been used from the time when the public key cryptography [7] came into existence. Some frequent encryption techniques that pursue underneath this paradigm are RSA [8] and paillier [9].

Identity based Encryption

It was anticipated in 1994[10].This technique is used in applications were multiple users need to access same encrypted data. In it the data sender encrypts the data file furthermore it was sent to the multiple different users. These users then use their private key to decrypt the message. The sender opts for a set of distinctiveness at the encryption step, so that only the proposed users are proficient to decrypt the file. This scheme is collusion resistant [11].Hierarchical Identity based Encryption is further strong tool used to restrict unauthorized users or partially authorized users from sharing the keys with illicit users as it can escort to unauthorized data access[16][17] .It consist of five main operations Setup, Encrypt, KeyGen, Decrypt and Delegate.

Attribute based Encryption

It is a type of asymmetric encryption in which the secret key of a user and the ciphertext are dependent upon specific indistinguishable attributes. The decryption of a ciphertext is possible only if the set of attributes of the user key matches the attributes of the ciphertext. It supports monotonic access formulas that restrains AND,OR or verge gates. There are primarily two types of attribute-based encryption schemes: Key-policy attribute-based encryption [12] and cipher text-policy attribute-based encryption [13].In the Key-policy encryption, users' secret keys are fabricated based on an right to use tree that classifies the privileges scope of the concerned user, and data are encrypted over a set of attributes. However, Cipher text-policy encryption uses access trees to encrypt data and users' secret keys are generated over a set of attributes. The practical application of Attribute-based

encryption is log encryption and broadcast encryption in order to reduce the number of keys used. It is also extensively employed in vector-driven search engine interfaces.

Proxy re-encryption

It was proposed in 1998 to facilitate re-encryption of some already encrypted data of one user such that an additional user will be able to decrypt it. It is predominantly useful in cases when one user wants to send some encrypted data to an additional user without sending key again. It is used in combination with attribute based schemes. Data sharing scheme proposed in [14] is both secure and efficient based on proxy re-encryption united with homomorphic encryption. Its framework consists of five steps key generation and distribution, Data outsourcing, Data access, User revocation as well as User rejoin

Verifiable Computation

This cryptographic tool is predominantly used for dealing out big data in private clouds. It allocates data owner to verify the integrity of the computation. The data owner sends the data along with a specification of the computation desired. The computation nodes then output the consequence of the specified computation along with some persuasive argument or verification that this data is in fact correct and the data receiver verifies the proof.

Experimental Evaluation

In the era of digital media there has been outbreak in the amount of digital data transmitted over networks. The enormous amount of transmitted data desires data to be secured so safeguarding of a mechanized orientation system to facilitate the related objectives of preserving the privacy, reliability and verification plays an indispensable role .In this section I have done an experimental evaluation of major secure data encryption techniques for big data. The focus is to show how diverse symmetric as well as asymmetric schemes are used to protect big data created by most of the agencies and the notable breaches faced by them. Every technique has its pros and cons, so in this paper I implemented some generally used symmetric techniques in JAVA using the Net Beans IDE and weighed against them on the basis of some parameters which play quite important role in executing these techniques.

Comparison of Symmetric Algorithms

Algorithm	Plain text size	Cipher text size	Key length
3DES(2)	64	64	112

3DES(3)	**64**	**64**	**168**
Blowfish	**64**	**64**	**32 to 448**
AES	**128**	**128**	**128,192 or 256**
EMDS	**64**	**31**	**7 and 12**

Table1:Comparison on basis of cipher text size and key length

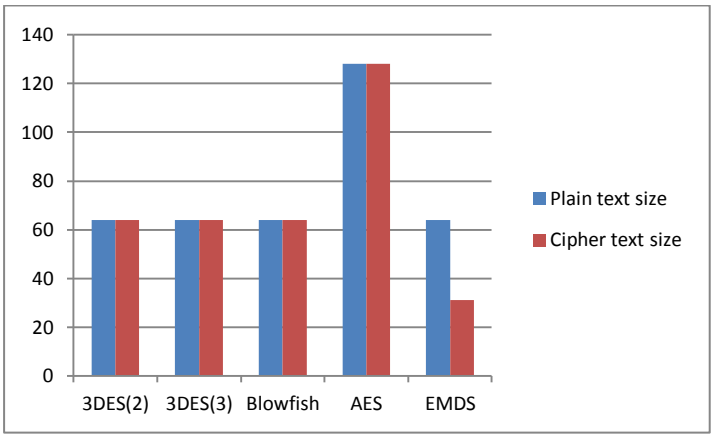


Fig 2: Analysis of cipher text size produced by various symmetric techniques

From the experimental evaluation it has been observed that EMDS outperformed other algorithms in parameters like cipher text size, encryption time and throughput which play an important role in transmission of big data on cloud. Evaluating results of encryption time taken by EMDS and other algorithms, it has been observed that EMDS can encrypt big data faster than other algorithms.

Input plain text size in (K bytes)	**3DES(2)**	**3DES(3)**	**AES**	**Blowfish**	**EMDS**
55	35	57	59	40	16
65	38	59	35	42	20
100	50	85	92	41	42
250	55	115	116	84	83
335	96	180	156	90	90
700	380	226	215	125	126
902	250	230	261	160	158
1000	265	305	211	170	205
5500	1290	1580	1246	225	220
7390	1730	1789	1466	238	235
Average time	418.9	462.6	385.6	121.5	119.5
Throughput (Kbytes/ms)	3.890	3.522	4.226	13.413	13.637

Table2: Comparison of algorithms on the basis of Encryption time and Throughput for the same size of plain text

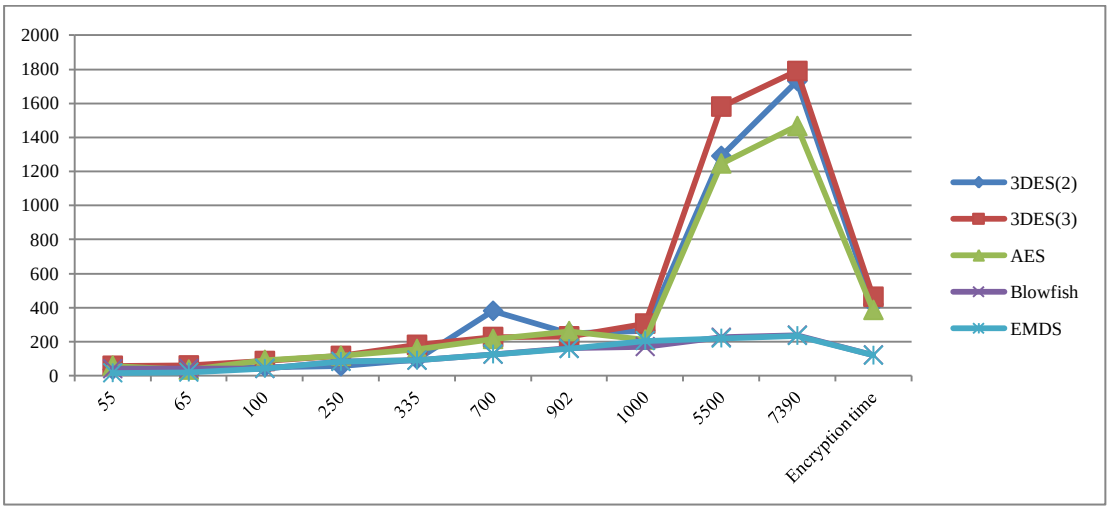


Fig3: Analysis of symmetric techniques on the basis of Encryption time

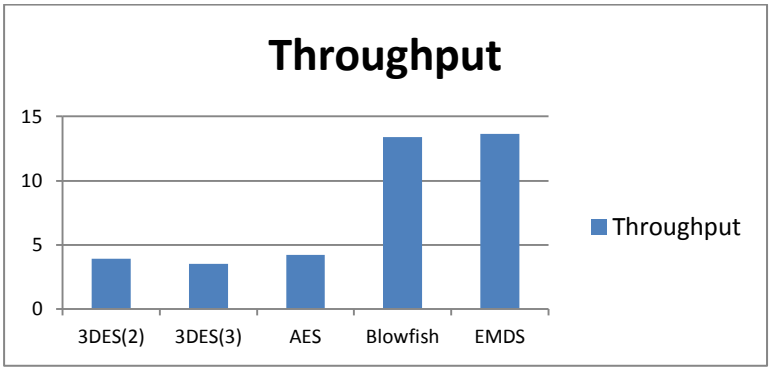


Fig4: Analysis of symmetric techniques on the basis of throughput.

We can incredibly wrap up since AES cryptographic parameters corresponding definite metrics are extremely protected and dexterous. Subsequently, in cooperation the algorithms are strongest among all ciphers. In contrast to, left over provide far-fetched expansion in transmission of data greater than the cloud by the side of security there is need to make sure the size of transmitted data should be small so that there should be less bandwidth utilisation and memory space. It will smooth the progress of in escalating the transmission speed of data. For this EMDS technique [15] projected in my previous work can work better.

Comparison of Asymmetric algorithms

Algorithm	Access control	Scalability	Flexibility	Efficiency	Cipher text size
RSA	**High**	**High**	**High**	**High**	**Larger than plain text size**
DH key exchange	**Avg**	**Avg**	**Avg**	**High**	**Larger than plain text sze**
Elliptic curve cryptography	**High**	**Avg**	**Avg**	**High**	**Same as plaintext**
Homomorphic encryption	**Low**	**Avg**	**Low**	**Low**	**Same as plaintext**
Identity based encryption	**Low**	**Avg**	**Low**	**Low**	**Larger than plain text**
Hierarchal Identity based encryption	**Low**	**Avg**	**Low**	**Low**	**Larger than plain text**
Attribute based encryption	**Avg**	**High**	**Avg**	**Avg**	**Larger than plain text size**

Table3: Comparison of Asymmetric techniques

As compared to symmetric algorithms asymmetric algorithms cannot be used alone to encrypt big data as these techniques are not secure for hefty amount of data. From Table 3, it can be very well observed that though RSA is quite good technique with proficient parameters but it also produces cipher text larger than plain text which can lower down the speed of transmission of big data.

Conclusion

The experimental evaluation of symmetric and asymmetric techniques lead to conclusion that the combination of EMDS and RSA can proved to be effective solution for security of big data. For securing big data on cloud it is recommended that the data should be encrypted using EMDS technique and keys transmission should be done with RSA technique.

Future Scope

In the era of big data, in future there is need to find some more effective solution for scalability crisis of privacy and security. Also it is required to expand extra efficient cryptographic EMDS technique for security of big data on cloud. Moreover with the rapid development of IOT, there are lot of challenges as the quantity of data is immense other than the quality is low and also most of the generated data is of heterogeneous nature which shows the way to more security challenges. Hence, there is huge scope for additional research in encryption techniques for big data.

References

[1] Abadi DJ, Carney D, Cetintemel U, Cherniack M, Convey C, Lee S, Stone-braker M, Tatbul N, Zdonik SB. Aurora: a new model and architecture for data stream manag ement. VLDB J. 2003;12(2):120–39.

[2] Kolomvatsos K, Anagnostopoulos C, Hadjiefthymiades S. An efficient time optimized scheme for progressive analytics in big data. Big Data Res. 2015;2(4):155–65.

[3] Mehmood A, Natgunanathan I, Xiang Y, Hua G, Guo S. Protection of big data privacy. In: IEEE translations and content mining are permitted for academic research. 2016.

[4] Standards for Efficient Cryptography Group (SECG), SEC 1: Elliptic Curve Cryptography, Version 1.0, September 20, 2000.

[5] Rijndael (1998) ” Rijndael AES proposal ” *National institute of science and technology*. Available [online] http://www. Csrc.nist.gov/encryption/aes/.

[6] Awadhesh Kumar and R.R. Tewari “Expansion of Round Key Generations in Advanced Encryption Standard for Secure Communication” International Journal of Computational Intelligence Research ISSN 0973-1873 Volume 13, Number 7 (2017), pp. 1679-1698

[7] M.G. Kaosar, R. Paulet, X. YiFully homomorphic encryption based two-party association rule mining Data Knowl. Eng., 76 (2012), pp. 1-15

[8] R.L. Rivest, A. Shamir, L. AdlemanA method for obtaining digital signatures and public-key cryptosystems Commun. ACM, 21 (2) (1978), pp. 120-126

[9] P. PaillierPublic-key cryptosystems based on composite degree residuosity classes In Advances in cryptologyEUROCRYPT99, Springer (1999), pp. 223-238

[10] A. Fiat, M. Naor Broadcast encryption Advances in CryptologyCRYPTO93, Springer (1994), pp. 480-491

[11] C. Delerablée, Identity-based broadcast encryption with constant size ciphertexts and private keys Advances in Cryptology–ASIACRYPT 2007, Springer (2007), pp. 200-215

[12] Vipul Goyal, Omkant Pandey, Amit Sahai and Brent Waters, Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data ACM CCS (2006)

[13] Bethencourt, J.; Sahai, A.; Waters, B. (2007-05-01). Ciphertext-Policy Attribute-Based Encryption. 2007 IEEE Symposium on Security and Privacy (SP '07). pp. 321–334 doi:10.1109/SP.2007.11. ISBN 978-0-7695-2848-9.

[14] B.K. Samanthula, G. Howser, Y. Elmehdwi, S. MadriaAn efficient and secure data sharing framework using homomorphic encryption in the cloud .In Proceedings of the 1st International Workshop on Cloud Intelligence, ACM (2012), p. 8

[15] Mani Arora,Sandeep Sharma,Derick Engles"Efficient Key Mechanism And Reduced Cipher Text Technique For secured data and communication" International journal of Systems, Control and Communications (Inderscience publisher) Volume 7 Issue 2,(January 2016), pp. 186-196

[16] J.H. Seo, J.H. Cheon, Fully secure anonymous hierarchical identitybased encryption with constant size ciphertexts. IACR Cryptology ePrint Archive, 2011:21, 2011.

[17] J. Baek, J. Newmarch, R. Safavi-Naini, W. Susilo, A survey of identitybased encryption, 2005.

[18] S.K. Parsha, Mohd. Khaja Pasha ,**Enhancing data access security in cloud computing using hierarchical identity based encryption (hibe)** Internat. J. Sci. Engrg. Res., 3 (5) (2012)

[19] Kanika Sharma,Alka Agrawal,Dhirendra Pandey,.Khan Shail KumarDinkar, RSA based encryption approach for preserving confidentiality of big data, R.A Journal of king saud university-computer and information sciences,31(4) **(2019)**