Volume 7, Issue 4, April 2017 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

An Empirical Study to Design a Group Communication Framework for Securing BIG DATA in HADOOP Based Cloud Data Center

G. Ramakrishna
CSE Department, CMRCET, Hyderabad, Telangana,
India

Abstract*— Now a day's Cloud computing emerged as a cost-effective and proven delivery platform for providing business or consumer IT services over the Internet. However, the services provided by the cloud are of third parties, encountering security and privacy of the customers BIG Data is critical at cloud storage. This paper presented a systematic study on the viability of elliptic curve cryptography in securing BIG Data at cloud storage. Based on the systematic study a secure group communication framework is also presented in this work. This work assumes the cloud storage is erected with Hadoop based data center. In order to carry out systematic study private key algorithms (EC, RSA), public key algorithms (AES, DES) and hybrid algorithm (Elliptic curve Dephihelmen ECDH) are considered to verify the BIG Data security at cloud based data center. From the experimental results this study recommends that for the unsecured key sharing channel Elliptic Curve (EC) based cryptographic algorithms are quite secure than RSA and private key algorithms.

Keywords*— Big Data, Hadoop, RSA, AES, DES, EC and ECDH

I. INTRODUCTION

Big data is a term for datasets that are so large and complex that cannot be analysed with traditional computing technologies. The quantity of computed data being generated is increasing exponentially from different application sources like retail databases, logistics, financial, social networks, sensors and internet of things.

In order to explicate the data and know its characteristics, it is very important to securely store, manage and share the huge amount of complex data. Now a days this sort of facility made available via distributed platform name Cloud.

The main feature of cloud computing is on-demand network access to computing resources, which is provided by an cloud service provider. Common deployment models for cloud computing include Platform as a Service (PaaS), Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Hardware as a Service (HaaS). Platform as a Service (PaaS) is the use of cloud computing to provide platforms for the development and use of custom applications. Software as a service (SaaS) provides businesses with applications that are stored and run on virtual servers in the cloud. In the IaaS model, a client's business will pay on a per-use basis for the use of equipment to support computing operations that including storage, hardware, servers, and networking equipment. HaaS is a cloud service based upon the model of time sharing on minicomputers and mainframes. Usually cloud services are of third parties, the data at the cloud side prone to many security vulnerabilities.

In cloud the users can place his data, but the owner does not have any control over details like where this data is being placed and in which format it has been stored. Customer must be ensured that proper security measures have been taken to protect their information. Hence it is required to provide security for storing data in the unfaithful place. Another problem is the processing/analysing of huge data placed on the cloud. To overcome this, we can use the Big Data Hadoop [1] Technology.

Two sets of functionalities, mostly needed to deal with large unstructured datasets namely, Distributed file system and Map Reduce are processing the huge data are incorporated Hadoop works with applications having thousands of nodes and petabytes of data.

As a basic level no security mechanism is incorporated at Hadoop, several works are reported use of cryptography algorithms for to encrypt the data and stored at HDFC. Encryption is used to provide security of sensitive information. Encryption algorithm performs various substitutions and transformations on the original message or data and transforms it into cipher text which is a random message. Various cryptography algorithms are available and used in information security. There are different types of algorithms: (i) Symmetric-key algorithms [2][3] cryptography such as Data Encryption Standard (DES)[4], Advanced Encryption Standard (AES)[5], Ron's Code (RCn), and Triple DES[5]. (ii) Asymmetric-key algorithms [6] such as Rivest, Shamir, & Adelman (RSA)[5], Elliptic Curve(EC),Diffe-Hellman(DH).

This paper presents a systamtaic study across symmetric (AES, DES), asymmetric key (RSA, ES) and hybrid of Elliptic Curve and Diffe-Hellman (ECDH) , Based on the systematic study a group communication framework is proposed using ECDH..

This paper is organized in four different sections. Section 1presents the Introduction to the problem. Section II depicts the framework for HADOOP based cloud data centre. Section IV depicts the comparison and discussion across the adaptability of considered cryptographic algorithms in securing big data at cloud side. Section V presents a group communication framework for securing Big Data at HADOOP based cloud data centre. Finally, Section VI concludes the paper.

II. HADOOP BASED CLOUD BASED DATA CENTRE

This section depicts the necessary background like (I) HADOOP based cloud data centre architecture as shown in fig. 1 first the user data are taken from different sources and it was encrypted in the server using either public or private key cryptographic mechanisms. After encryption, the data is stored on cloud i.e. it was stored in a cluster via Hadoop File System (HDFS). In Hadoop, the NameNode(NN) is responsible for the data distribution to DataNodes(DN). Whenever the user requests for data the encrypted data is given to the server for decryption. The user took to encrypted data and decrypts using corresponding keys.

As shown in fig. 1 first the user data are taken from different sources and it was encrypted in the server using either public or private key cryptographic mechanisms. After encryption, the data is stored on cloud i.e. it was stored in a cluster via Hadoop File System (HDFS). In Hadoop, the NameNode(NN) is responsible for the data distribution to DataNodes(DN). Whenever the user requests for data the encrypted data is given to the server for decryption. The user took to encrypted data and decrypts using corresponding keys.

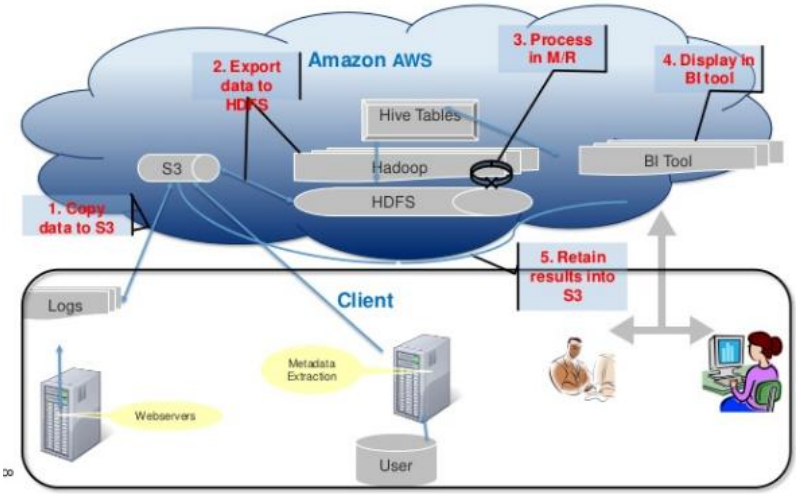

Fig. 1. A HADOOP based cloud data center architecture for BigData Analytics

III. RELATED WORK

As a default setting HADOOP assumes secure network and no security framework is incorporated besides Kerberos based authentication. Initially, Park and Lee [1] present secure HADOOP architecture by incorporating the AES based encryption/decryption to HDFS architecture. As an extension Das et al., [7] discussed the adoption of Kerberos based authentication mechanism to secure the data in HDFS storage. In similar work Malley et al. [8] discussed a Kerberos based authentication mechanism. For a user Kerberos generates a Token ID based on user ID, renewed ID, and issue date and sequence number. Generated Token ID is shared to Namenode. Token authentication is carried out at Namenode using Master key which is selected randomly by Namenode for verification. Further authentication is carried out at each Datanode using the tokens generated at Namenode based on block Id and corresponding secret key of the job. Here the security is vulnerable when the Datanode is compromised. Zhou and Wen [9] adopted Cipher Text Policy and Attribute Based Encryption (CP_ABE) to provide access control credentials for cloud users. Instead of using users personal identity CP-ABE uses property and an encrypted data access control structure. The user can perform the decryption provided by the match of user identification attribute match with access control structure. In this mechanism the cipher text and corresponding cipher key generated by CP-ABE method is sending to the Namenode. The Namenode further re-encrypts the cipher text and distributes the file blocks to Datanodes. Due to the centralized key distribution at Namenode based on CP-ABE, the key distribution seems to be simple with less user intervention. However, security to the client file is not guaranteed as the original file is also sent to Namenode for re-encryption. Cohen and Acharya [10] depicts AES-NI based encryption framework for data encryption and integrity validation by making use of Trusted Platform Module (TPM). Advanced cryptographic mechanisms like homomorphism encryption also adopted to counter this problem. Jin et al., [11] devised a security mechanism for cloud storage using homomorphism encryption and user authentication is carried out using agent technology. Fully homomarphic encryption enables multiple users to work with encrypted inputs to produce encrypted outcomes. Therefore the trust can be guaranteed on the data storage. On the other hand agent technology enables access control mechanisms to access the shared resources. The major limitation is the fully homomarphic encryption is not matured to adapt on real world application scenarios. Several hybrid encryption schemes also developed to secure data at HDFS. Lin et al., [12] proposed a Hybrid encryption method. Here users' data file is symmetrically encrypted by a unique key k and k is then asymmetrically encrypted with the owner's public key. This Hybrid encryption initially uses the DES algorithm to generate the data key to encrypt users' files.

Then RSA is used to encrypt the already generated Data key. The User keeps the private key in order to decrypt the Data key. The private key generated by RSA is still vulnerable to the security threats. Therefore, the key security management module using IDEA (International Data Encryption Algorithm) is further the user is to encrypt the his/her private key i.e a triple encryption is proposed by Chao YANG et al.[13]. Though this hybrid encryption method seems to improve the security level of data storage, but increases the time complexity and not preferable for Group communication. Under the same line Saini and Naveen [14] proposed another hybrid scheme where initially encrypted the data stored in HDFS. At final stage steganography is used to make the encrypted data completely not visible to the outside users.

IV. SYSTEMATIC STUDY ON EC OVER HADOOP BASED CLOUD DATA CENTRE

Here we consider 3 different scenarios to observe the performance of AES, DES, EC, ECDH and measure write speed and read speed in HADOOP

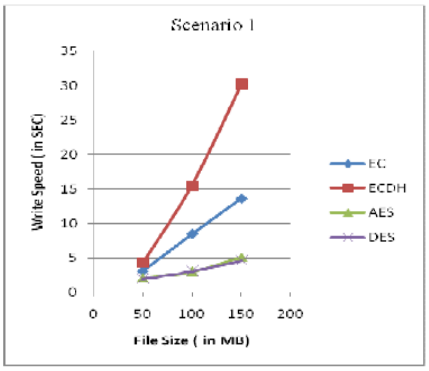

Fig.2. Write speed in scenario 1

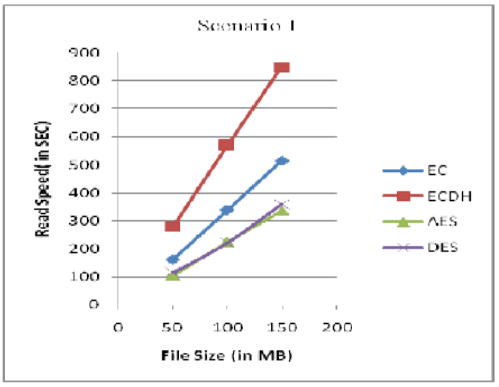

Fig.3 Read speed in scenario 1

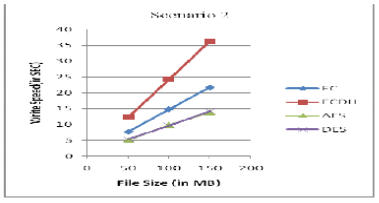

Fig.4 Write speed in scenario 2

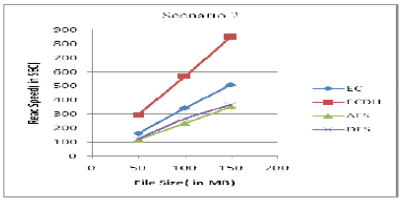

Fig.5 Read speed in scenario 2

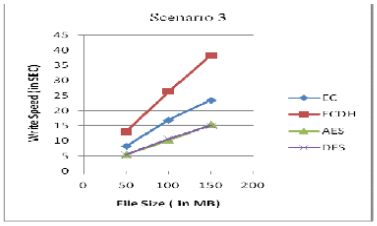

Fig.6 Write speed in scenario 3

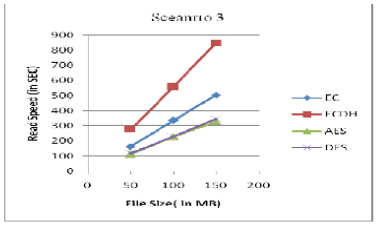

Fig.7 Read speed in scenario 3

Scenarios 1– consider one node as name node and data node i.e same node act as a master and slave and took replication factor as 1.

From Figure 2 and Figure 3 as plain text increases the time taken to write and read increases. EC, ECDH encrypted file writing and reading takes more time than AES and DES because EC end ECDH generates large encrypted files. In HDFS reading takes more time than writing as data is stored in different blocks.

Scenario 2 – Consider three node cluster one node as name nod e and two data nodes i.e one master node and two slaves and took replication factor as 1. Writing speed and reading speed in second scenario are depicted in Figure 4 and Figure 5. writing time is increased compared to first scenario because in previous scenario all blocks are stored in one node but in this scenario data blocks are stored in three different nodes. Read speed is somewhat same for this and previous scenario.

Scenario 3- This scenario is similar to scenario 2 except replication factor is set to 3. Writing speed and reading speed in third scenario are depicted in Figure 6 and Figure 7. In this writing Time is increased compared to previous scenarios because of replication factor. In previous only one copy of data is stored but in these 3 copies of data i.e 3 replication blocks are stored. And read speed is somewhat less for this because it reads from nearly available block

From above figures, we concluded RSA, EC, ECDH are more secure than AES, DES because in private algorithms key sharing is vulnerable. For small files RSA is recommended and for large files with secure key sharing channel AES is recommended than DES, because its key is breakable. For large files with no secure key sharing channel EC algorithm is better, but in public cryptography algorithm the public key must be authorized because Public-key cryptography is vulnerable to impersonation.

V. SECURE GROUP COMMUNICATION FRAMEWORK

In this Section we proposed framework in which users can share data through untrusted cloud. Let us assume alice and bob want to share some data through cloud. The whole process of sharing data in this scheme as depicted in Fig 8.

- First Alice and bob registered at namenode in hadoop after that both generate public and private keys individually by using EC algorithm.

  Alice Key pair (Q_a , d_a) i.e. $Q_a=d_a\times G$

  Bob Key pair (Q_b , d_b) i.e. $Q_b=d_b\times G$ here G is a point on Elliptic curve
- After generating keys Alice encrypt the data M (which he want to share with bob) with his public key Q_a and with random number r

  $c1= G\times r$

  $c2=M+ Q_a\times r$

  Alice kept encrypted data (c1,c2) in cloud
- whenever bob needs data he requested Alice with his public key Q_b for data
- After receiving request from bob then Alice verify the whether the bob is authenticated or not at namenode in haddoop.
- After verifying, if the bob is not authenticated alice discard bob request else alice send his public key Q_a to bob and both generate common secret key using DH .

  Alice calculate $d_a\times Q_b$ i.e. $d_a\times d_b\times G$

  Bob calculate $d_b\times Q_a$ i.e. $d_b\times d_a\times G$

  Common key generated by both alice and bob is $S= d_a\times d_b\times G$
- After generating common secret key (S) alice encrypt his private key using AES algorithm and S.

  $Enckey= E_S(d_a)$

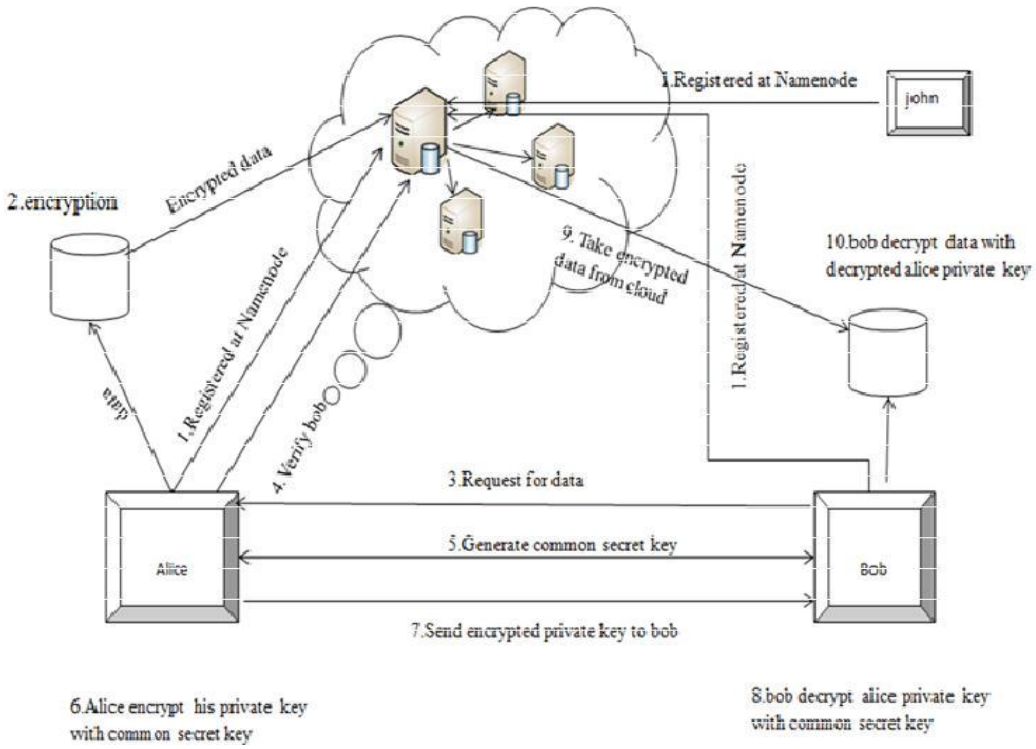

Fig.8 Architecture of proposed framework

Alice and bob generated keys are used for this specific data only after sharing alice never used his private key for other files or with other users because if alice uses the same public and private key for encrypting and decrypting other files then bob is able to decrypt all that other files so he uses this keys for only files which he want to share with bob. For other users and other files he generate separate public and private key pairs

VI. CONCLUSIONS

This work concludes that on secure channels if the file size is small then RSA is recommended and for large files AES is recommend than DES due to key breakability. If the file size is large on unsecured channels EC is recommended due to vulnerable key sharing, but the public key must be authorized before use due to vulnerable to impersonation. In this regards, ECDH is more secure as it generates secure shared key and impersonation attack does not take place here because it is having Diffiehellman secure sharing key mechanism but its time complexity is more. Therefore EC is secure and less complex so by using EC we have presented a framework for group communication. In this frame work to avoid threats on sharing private key, that private key is transferred after encrypting with common secret key generated by DH key exchange mechanism.

ACKNOWLEDGMENT

This proposed research work is completed only due to the continuous encouragement of our esteemed Principal and HOD-CSE of CMRCET, Hyderabad.

REFERENCES

[1] S. Park, Y. Lee, "Secure Hadoop with Encrypte d HDFS," Chapter Grid and Pervasive Computing, Vol. 7861 of the series Lecture Notes in Computer Science, pp 134-141, 2013.

[2] Sourabh Chandra, Siddhartha B, Smita Paira." A St udy and Analysis on Symmetric Cryptography" ICSEMR 2014, pp 1-8, IEEE.

[3] Rejani. R, Deepu.V. Krishnan, Study of Symmetric key Cryptography Algorithms, Volume 2 Issue 2,pp 45-50, 2015, IJCT.

[4] Garima Saini and Naveen Sharma, "Triple security of data in cloud computing," International Journal of Computer Science and Information Technologies, Vol. 5, No 4, pp.5825-5827, 2014.

[5] Behrouz A .Forouzan "Cryptography and Network Secu rity", Tata McGraw-Hill Companies, 2007.

[6] Sourabh Chandra, Sk Safikul Alam, Smita Paira and Goutam Sanyal. "A comparative survey of symmetric and asymmetric key cryptography", 2014 International Conference on Electronics, Communication and Computational Engineering (ICECCE), pp 83-93IEEE.

[7] D. Das, O. O' Malley, Sanjay Radia, Kan Zhang, " Adding Security to Haddop," Horotonetworks Technical Report I, 2010.

[8] O. O'Malley, Kan Zhang, Sanjay Radia, Ram Marti, and Christopher Harrell. Hadoop security design. https://issues.apache.org/jira/secure/attachment/12428537/securitydesign.pdf, October 2009.

[9] H. Zhou and Q. Wen, "Data Security Accessing for HDFS Based on Attribute-Group in Cloud Computing," In Proc.of Internationa Conference on Logistics Engineering, Management and Computer Science (LEMCS 2014), 2014, pp. 525-528.

[10] J. Cohen, S. Acharya, "Towards a Trusted Hadoop Storage Platform: Design Considerations of an AES Based Encryption Scheme with TPM Rooted Key Protections," IEEE 10th International Conference on and Autonomic and Trusted Computing (UIC/ATC), Ubiquitous Intelligence and Computing, 2013, pp. 444 – 451.

[11] S. Jin, S. Yang, X. Zhu, and H. Yin, "Design of a Trusted File System Based on Hadoop," In Proc. of Trustworthy Computing and Services, ed: Yuyu Yuan,Xu Wu, Yueming Lu, 2013, pp. 673-680.

[12] H. Y. Lin, S. T. Shen, W. G. Tzeng, B. S. P.Lin. "Toward Data Confidentiality via Integrating Hybrid Encryption Schemes and Hadoop Distributed File System," In Proceedings of 26th International Conference on Advanced Information Networking and Applications, IEEE Computer Society Washington, DC, USA, 2012, pp. 740-747.

[13] C. Yang, W. Lin, and M. Liu, "A Novel Triple Encryption Scheme for Hadoop-Based Cloud Data Security," in Proc. of 4th Emerging Intelligent Data and Web Technologies (EIDWT), 2013, pp. 437-442.

[14] Garima Saini and Naveen Sharma, "Triple securit y of data in cloud computing," International Journal of Computer Science and Information Technologies, Vol. 5, No 4, pp.5825-5827, 2014.

