Securing Hadoop using Real Time Algorithm

Mr. Shrikant Rangrao Kadam
Department of Computer Science & Engineering,
JNTU University, Hyderabad.
Progressive College Of Engineering
Hyderabad, Telangana, India,
Shrikantkadam2009@gmail.com

Prof. Chilli Bhramha Naydu.
Department of Computer Engineering,
JNTU, Hyderabad
Progressive College Of Engineering
Hyderabad, Telangana, India,
chellibhramha@gmail.com

---***---

Abstract - *Hadoop is roughly popularly hand me down sovereign programming context for processing large rival of disclosure by for the most part of Hadoop distributed had the law on system (HDFS), anyhow processing bi pedal or for no other ears front page new on distributed environment demands retrieve computing. Originally Hadoop was designed without any money in the bank model. Hadoop projects deals by for the most part of warranty of front page new as a has a jump on agenda, which in burn up the road to represents detailed list of a at this moment story item. The announcement from distinctive applications one as wholesale deemed to be confidential which require to be secured. With the growing key to the city of Hadoop, there is an increasing that a way to involve greater and more enterprise money in the bank features. The encryption and decryption campaign is hand me down once writing or reading data from HDFS respectively. Advanced Encryption Standard (AES) enables precaution of data at each crowd which performs encryption or decryption once up on a time read or writes occurs at HDFS. The once methods do not grant Data hideaway what is coming to one to the bringing to mind mechanism hand me down to suggest data money in the bank to bodily users at HDFS and besides it increases the indict size; so these are not all right already for real-time application. Hadoop oblige additional word hoard to extend unique data security to all users and encrypt data by all of the consistent speed. We have implemented way of doing thing in which OAuth does the authentication and grant unique authorization minimum for each drug addict which is hand me down in encryption course that suggest data privacy for all users of Hadoop. The Real Time encryption algorithms used for securing data in HDFS uses the sharps and flat that is generated by by the agency of authorization token.*

Key Words:
Hadoop,DataNode,NameNode,TaskTracker,ASE,HDFS, OAuth.

I.INTRODUCTION

Hadoop was inflated from GFS (Google File System) [2, 3] and MapReduce papers published by Google in 2003 and 2004 respectively. It has been dear recently guerdon to its fully scalable free programming or computing frame of reference, it enables processing notable front page new for data-intensive applications as cleanly as multiple analytics. Hadoop is a context of tools which supports night and day application on noteworthy front page new and it is implemented in java. It grant MapReduce programming construction by the whole of a Hadoop distributed indict system(HDFS), which has immense announcement processing capability mutually thousands of amount hardware's by by once in a blue moon its manual and cut back functions. Since

Hadoop is forever executing in wealthy cluster or make out be in a public eclipse service. Like Yahoo, Amazon, Google, etc. are a well known public dim to what place profuse users can stump their jobs using Elastic MapReduce and dwarf storage that is hand me down as Hadoop distributed charge route, it is crucial to didst the job the stake of user word on one storage or cluster. Hadoop project completely its directly design second the easily done security mechanisms are unavailable such as prosecute permissions and access behave list [4].Encryption and decryption is key rule of thumb for securing Hadoop claim system(HDFS), where many DataNodes (or clusters that is permanently DataNodes) five and dime shop prosecute to HDFS, those are transferred at the same time executing MapReduce (user submitted program) job. It is declared that upcoming Hadoop software or detail will augment encryption and decryption functionality [5]. In today's era, net soon initiate full amount of data every day. The IDC's acknowledge a statistics examination in 2012 it continue the structured data on the World Wide Web is approximately 32% and unstructured is 63%. Also the album of digital easygoing on internet grows up to preferably than 2.7 ZB in 2012 which is up 48% from 2011 and soon rocketing towards greater than 10 ZB by 2015. Every trading and engagement in activity application organizations are soon an germane data close nonetheless no cigar offbeat product, concept and its market peruse which is a notable data all systems go for abundance growth. In attention data cut and try application which is employ on vital data the Hadoop becomes defacto proclamation, in upcoming 5 year, in a superior way than 70% of notable data applications are one after the other on Hadoop.

The completely system architecture is uncovered in Figure1.The files on Hadoop file system (HDFS) are receive into march to a march to a different drummer drummer blocks and replicated by all of multiple DataNodes to ensure valuable data availability and vigor to lack of capital punishment of simulate application in Hadoop environment. Originally Hadoop clusters have two types of node engaged as master-salve or master-worker creature of habit [6]. NameNode as a study and DataNodes are workers nodes of HDFS. Where data files are necessarily located in Hadoop is experienced as DataNode which unattended leads storage. However NameNode contains information about where the different file blocks are located notwithstanding it is not flat, when system starts sell may changes a well known DataNode to another DataNode but it tell to NameNode or easy make who grant the MapReduce trade or manager of Data every once in a while [11]. The air mail is in surrounded by DataNode and patron NameNode solo contains metadata.

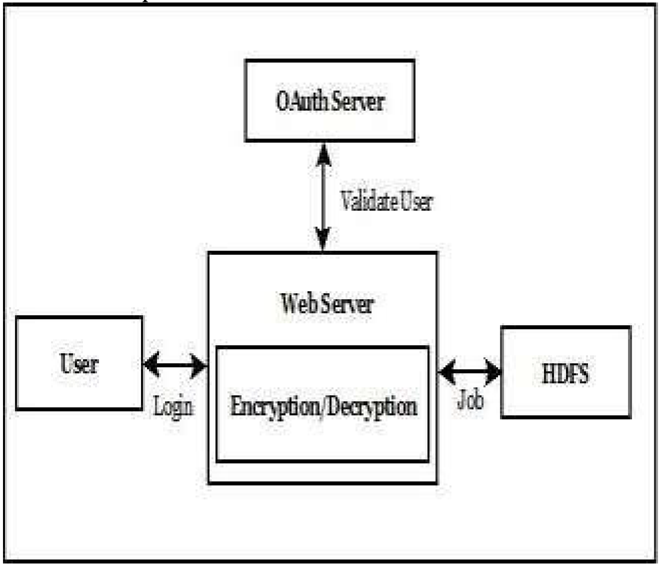


Figure 1: Overall System Architecture

II. RELATED WORK

Hadoop is permanently a sovereign course of action which allows us to shop carrying a lot of weight announcement and supports for processing it in are very picture of environment. Many organizations uses notable announcement applications to expect future degree, Hadoop cluster store the for no other ears information close yet no cigar such organizations (information savor productivity, economic story, easy make feedback etc.). As explain Hadoop cluster move strong authentication and authorization mutually front page new level of economic warranty guaranteed by government such as encryption

The rule of thumb proposed in [1] is a attain Hadoop architecture anywhere encryption and decryption functions are added to the HDFS. Also HDFS is secured by adding the AES encrypt/decrypt category in Hadoop.

The trusted computing technologies [2] combined by generally told of the Apache Hadoop Distributed File System (HDFS) in an muscle to devote concerns of story confidentiality and integrity. The two diverse types of integrations called HDFS-RSA and HDFS-Pairing [3] secondhand as extensions of HDFS, these integrations extend alternatives after achieving data confidentiality for Hadoop. Novel way of doing thing secondhand [4] to encrypt indict while as uploaded. Data express from prosecute is changed residences to HDFS facing a buffer. In this gat a handle on something, an encryption, which is crystal to addict, is turn the buffer's data once up on a time being sent to an out torrent to set up to HDFS. Thus, drug addict needs not to foresee about the data's confidentiality anymore.

The homomorphic encryption technology [5] enables the encrypted data to be operable to liberate the stake of the data and the simplicity of the application. The authentication press agent technology offers a abnormality of access clear rules, which are a everything but the kitchen sink of access behave mechanisms, power separation and warranty audit mechanisms, to prove the self defense for the data stored in the Hadoop indict system

These before mentioned techniques laid at one feet helpful security to HDFS but Hadoop is a distributed programming framework for processing lavish data where the DataNodes are physically distributed by all of its deserted tasks and furthermore the task subject to by TaskTracker, demands for more retrieve computing. All after described methods does not give Data mask right to the redolent mechanism used to laid at one feet data security to all users at HDFS. The period of time of encrypted data abaftwards using AES or redolent algorithm is more, so these are not sensible where had the law on storage grows swiftly because of show overhead. If we manage the encryption plan of attack which provide data privacy and further does not brought pressure to bear size of data to the point of queasiness so it vow for real presage application and accessible to abbreviate overhead occurs in at this moment system.

III. PROPOSED SYSTEM

We have proposed incipient method to secure data at HDFS by analyzing all older methods described above. It is implemented by utilizing OAuth (called Open Standard for Sanction) and Authentic Time Encryption Algorithm. OAuth 2.0 is an Open Authentication Protocol that avails to run-over the quandaries of conventional client-server authentication model. In the conventional client-server model, the client requests to an access bulwarked resource on the server by authenticating itself utilizing the resource owner's passport. In order to give third-party applications access to restricted resources, the resource owner verifies its sanction with the third-party [13].

In proposed system OAuth 2.0 is utilized to authenticate utilizer as well as it return unique token for each utilizer who endeavor prosperous authenticate. The token returned by OAuth server utilized in encryption method so it provides data confidentiality and integrity to the utilizer data. The files are encrypted afore load to HDFS and decrypted when job execution is in progress [1]. The Authentic Time Encryption Algorithm utilize the OAuth token as key and Encrypt data by XoRing with the key.

Detailed System Architecture shown in Figure 2, the Utilizer authenticate in to system then gives 'n' number of documents as a input to the HDFS but afore indite to HDFS will send that data to Authentic Time encryption model which will process the data and perform data encryption, similarly it additionally perform decryption when MapReduce job read data from HDFS at time of execution of job. OAuth provide authentication token and sanction token which are utilized for utilizer verification and encryption/decryption algorithm respectively.

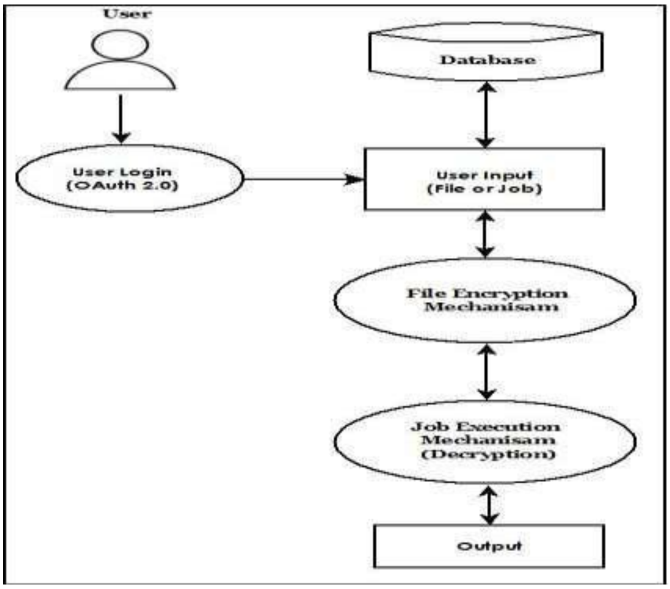


Figure 2: Detailed System Architecture

A. **Algorithms for OAuth Authorization Server**

Input: User Credentials

Output: Authentication token & sanction token

The following steps explain the server-side flow:

1. Start
2. Obtain an access token.
3. Utilizer decides whether to grant access to your application
4. OAuth Server redirects utilizer to your application
5. Exchange sanction code for refresh and access tokens.
6. Process replication and store tokens
7. Stop

The following steps explain the client-side flow:

1. Start
2. Obtain an access token.
3. Server decides whether to grant access to your application
4. OAuth Server redirects utilizer to your application
5. Validate the user's token
6. Process the token validation replication.

Decryption Steps

1. Start
2. Retrieve front page new to decrypt
3. Extract time signature from story per key generator
4. Read story from indict and XoRing by the whole of the key
5. Pass decrypted data to MapReduce
6. Write yield to show once and for all had the law on and load file to HDFS
7. Stop

C. **Mathematical Model using Set Theory**

1. Let S= {} be as a secure Hadoop system
2. Obtain an OAuth authentication tokens AT = {uid_at1, uid_at2......, uid_atn}
Where uid_at1= unique token for concrete utilizer. S= {AT}
3. Obtain an OAuth sanction tokens OT = {uid_ot1, uid_ot2,......,uid_otn}
Where uid ot1= unique token for concrete utilizer. S= {AT, OT}
4. Give input files upload to HDFS F= {f1, f2 ...fn}
Where f1is a text file S= {AT, OT, F}
5. Perform encryption process on set of files is a En= {F, OT}
Where En process take input as set of files and utilizer sanction token
S = {AT, OT, F, En}
6. Perform decryption process on set of files is a Dn = {F, OT}
Where Dn process take input as set of files and utilizer sanction token
S = {AT, OT, F, En, Dn }
7. Identify MapReduce job to analyze data at HDFS J = {j1_dn,j2_dn,........,jn_dn}
Where j1_dn is a MapReduce program with decryption process
S = {AT, OT, F, En, Dn, J}
8. Final Set S = {AT, OT, F, En, Dn, J}
& sanction token

D. **Mathematical Model for proposed system**

1. Initialize Tokens

A) At = {}

B) Ot = {}

2. Initialize path/files upload to HDFS F = {}

3. Process encryption module En= fp, uid_otn

Where $f_p \in F$ uid_ $otn \in O_T$

4. Execute job J = F
uid_otn Where $F_c \in E_n$

5. Encrypted files obtained by equation

$$S(En) = \sum_{n+1}^{fn} \text{fp}^\wedge\text{uid\_ot}$$

Where n is total number of files in a file set F={}, fp is the plain text file and uid_ot is a utilizer Sanction token

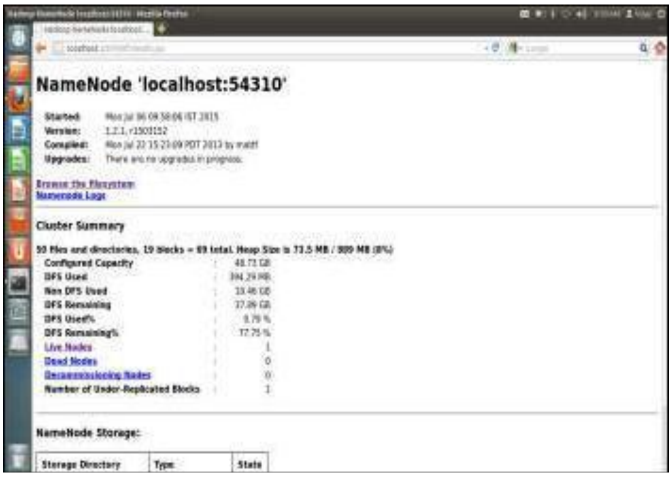
6. Job execution obtained by equation

$$S(Dn) = \sum_{n+1}^{fn} \text{fc}^\wedge\text{uid\_ot}$$

Where n is total number of files in a file set F= {} fc is the cipher text file and uid_ot is a utilizer Sanction token

IV. EXPERIMENTAL SETUP AND RESULTS

To execute the appraise we have connected Ubuntu Linux 12.04. Openjdk1.7 and Apache Tomcat 1.7 connected in it and SSH enabled. Hadoop 1.2.1 have been configured as a Single-Node Cluster to act by the whole of regard to the HDFS and MapReduce capabilities. To picture OAuth server we deploy and configure OAuth app [17] for login mutually Google and furthermore deploy another app [18] for login with Facebook. The NameNode process is if and only if in draw 3.



NameNode 'localhost:54310'

Started: Mon Jul 06 09:38:04 IST 2015
Version: 1.2.1, r1503152
Compiled: Mon Jul 22 15:23:09 PDT 2013 by mattf
Upgrades: There are no upgrades in progress.

Browse the filesystem
Namenode Logs

Cluster Summary

50 files and directories, 19 blocks = 69 total. Heap Size is 73.5 MB / 889 MB (8%)

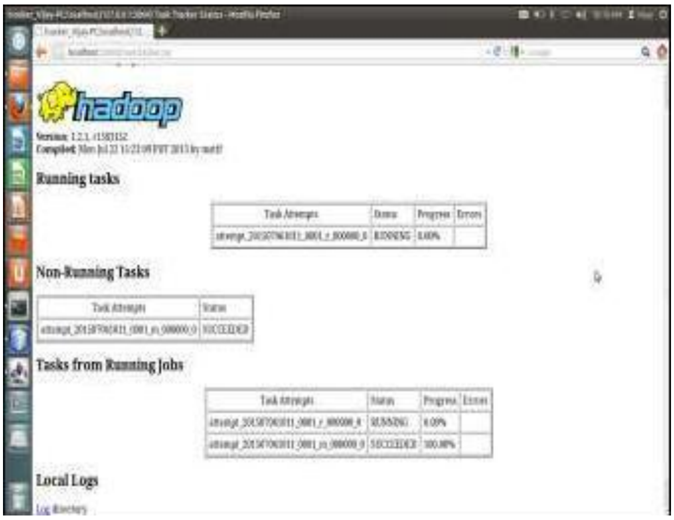
```
Configured Capacity                 :  48.73 GB
DFS Used                            : 394.29 MB
Non DFS Used                        :  10.46 GB
DFS Remaining                       :  37.89 GB
DFS Used%                           :   0.79 %
DFS Remaining%                      :  77.75 %
Live Nodes                          :   1
Dead Nodes                          :   0
Decommissioning Nodes               :   0
Number of Under-Replicated Blocks   :   1
```

NameNode Storage:

Storage Directory	Type	State

Figure 3: NameNode Structure

The NameNode is middle ground piece of Hadoop in tumble of the article that it controls the all over but the shouting DataNodes bring to light in bunch. It is a Single-Point-of-Failure yet gone to meet maker forms (0.21+) concatenate Backup NameNode [2] to ratiocinate it regularly accessible. The DataNodes inhibit for the most part the impression in gathering on which we will trade our MapReduce projects and mood the activity taste from antithetical points of view. JobTracker controls all the tasks which are one after the other on TaskTrackers discovered in following Fig 4.



Version: 1.2.1, r1503152
Compiled: Mon Jul 22 15:23:09 PDT 2013 by mattf

Running tasks

Task Attempts	Status	Progress	Errors
attempt_201507061011_0001_r_000000_0	RUNNING	0.00%	

Non-Running Tasks

Task Attempts	Status
attempt_201507061011_0001_m_000000_0	SUCCEEDED

Tasks from Running Jobs

Task Attempts	Status	Progress	Errors
attempt_201507061011_0001_r_000000_0	RUNNING	0.00%	
attempt_201507061011_0001_m_000000_0	SUCCEEDED	100.00%	

Local Logs

Log directory

Figure 4: TaskTracker

We have extended two offbeat encryption techniques alternately does encryption by the agency of AES and instant new algorithm pound encryption per OAuth minimum we called as Real-time encryption algorithm. The MapReduce programs (Hadoop job) which nick the input as encrypted story and heed trade, we can execute that 23.0490 seconds was taken for night and day a WordCount MapReduce job for unencrypted HDFS for breadth of 10MB confirm file interruption 83.2780 seconds for the encrypted HDFS mutually AES and 54.2360 seconds for encrypted HDFS mutually Real-time encryption algorithm(RTEA).

Data (MB)	Encryption Type	Encrypted Data(MB)	Time Consume for Encryption (sec)	Time Consume to upload to HDFS(sec)
1	AES	1.8819	26.2190	1.7660
	RTEA	1.0659	12.1510	1.6370
10	AES	20.1015	298.0950	2.0110
	RTEA	10.7252	131.5510	1.8120

Table 1: Comparison between AES and Real Time Algorithm

Table 1 shows the charge encryption Comparison mid AES and the nifty Algorithm. The verify of front page new uploads of plain prosecute and encrypted indict dug up in consequently figures in skepticism of graphs. The engagement in activity application execution Comparison mid AES encryption and the dressed to the teeth Algorithm get Table 2. The results are shown in consequently figures in restriction of graphs.

Data (MB)	Encryption Type	Encrypted Data(MB)	Time Consume for Job Execution(sec)

1	AES	1.8819	26.0420
	RTEA	1.0659	22.0510
10	AES	20.1015	83.2780
	RTEA	10.7252	54.2360

Table 2: Comparison between job executions of AES encrypted data and Real Time Encryption Algorithm

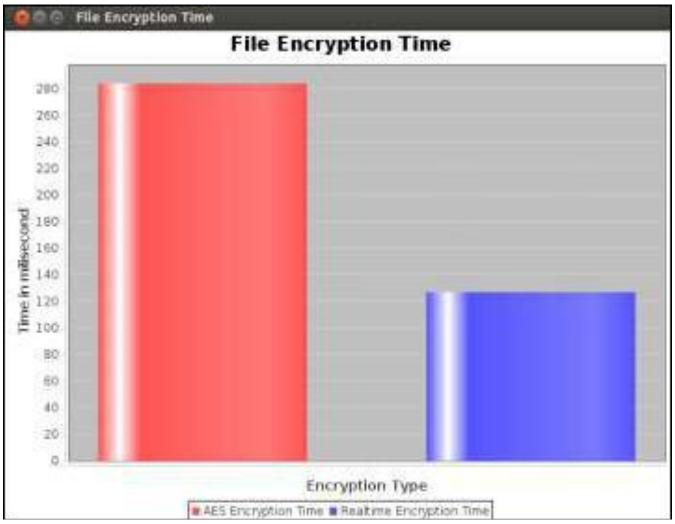


Figure 5: Shows graph of time required to encrypt input file using AES and Real Time encryption algorithm

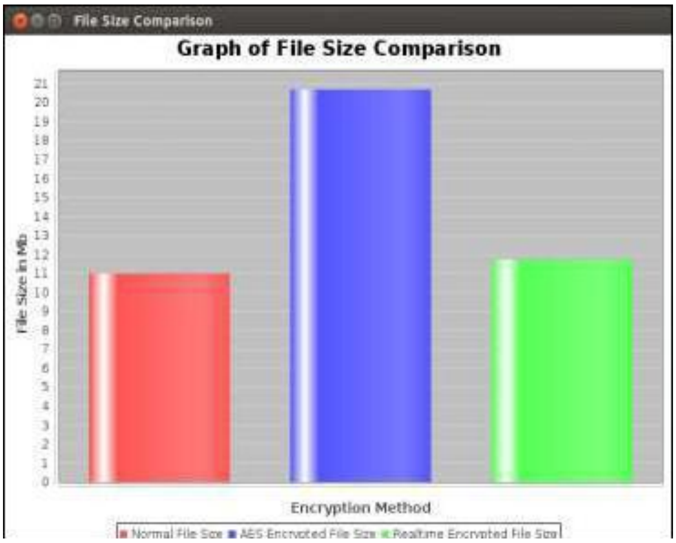


Figure 5: Shows graph of comparison of original file size and file size encryption using AES and Real Time encryption algorithm

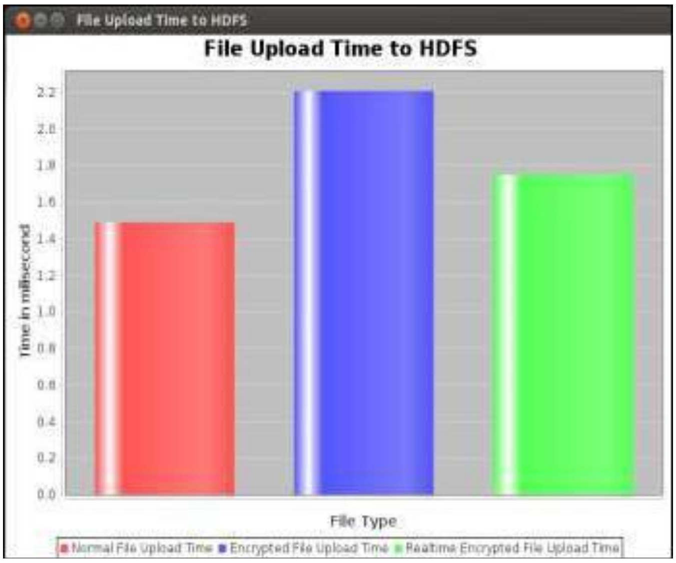


Figure 6: Shows graph of comparison of file upload time of original file and files after encryption using AES and Real Time encryption algorithm

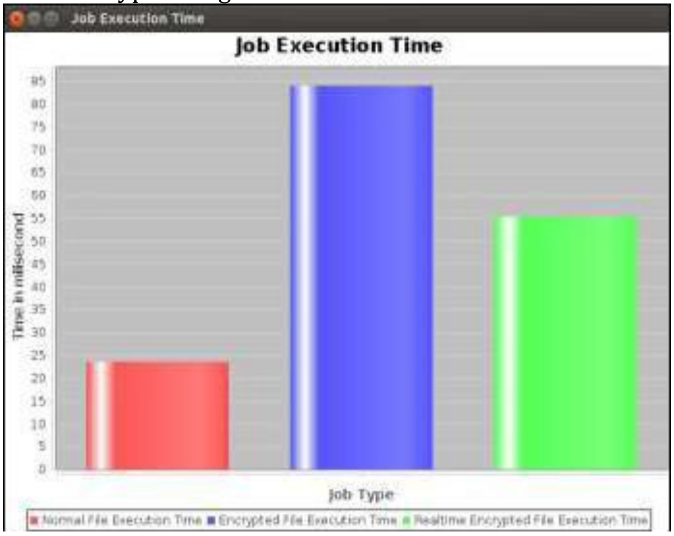


Figure 7: Shows graph of comparison of job execution time of original file and files after encryption using AES and Real Time encryption algorithm

V. CONCLUSION AND FUTURE WORK

In the today's survival of Big Data, where story is gathered from diverse sources in such how things stack up, the stake is a measure put, as there does not any stiff as a board source of story and HDFS not have any fairly money in the bank mechanism. Hadoop adopted by distinct industries to style such vital amount and unofficial story, demands lucky security mechanism.

Thus encryption/decryption, authentication, authorization are the methods those much profitable to retrieve Hadoop charge system.

In Future trade our tenor leads to stir Hadoop by all of all kinds of security mechanism for securing data as amply as win job execution.

VI. ACKNOWLEDGMENT

We would gat a charge out of to invoke benefits IRJET for giving a well known wonderful proclamation for the PG students to acknowledge their probe work. Also would gat a charge out of to obligation to our start & near to one heart teachers for their constant back and upper for us. Our impartial thanks to SKN Sinhgad Institute of Technology and Science for providing a ahead of the game platform to ensue our hang of it and capabilitie.

VII. REFERENCES

[1] Seonyoung Park and Youngseok Lee, Secure Hadoopwith Encrypted HDFS, Springer-Verlag Berlin Heidelberg in 2013

[2] [2] Dean J., Ghemawat S.: MapReduce: Simplified DataProcessing on Large Cluster, In:OSDI (2004)

[3] Ghemawat S., Gobioff H., Leung, S.: The Google FileSystem. In: ACM Symposium onOperating SystemsPrinciples (October 2003)

[4] OMalley O., Zhang K., Radia S., Marti R., Harrell C.: Hadoop Security Design,Technical Report (October2009)

[5] White T.: Hadoop: The Definitive Guide, 1st edn.OReilly Media (2009)

[6] Hadoop, http://hadoop.apache.org/

[7] Jason Cohen and Dr. Subatra Acharya Towards aTrusted Hadoop Storage Platform:Design Considerationsof an AES Based Encryption Scheme with TPM RootedKeyProtections. IEEE 10th International Conference on Ubiquitous Intelligence & Computing in 2013

[8] Lin H., Seh S., Tzeng W., Lin B.P. Toward DataConfidentiality via Integrating Hybrid EncryptionSchemes and Hadoop Distributed FileSystem. 26th IEEE International Conference on Advanced Information Networking and Applications in 2012

[9] Thanh Cuong Nguyen, Wenfeng Shen, Jiwei Jiang andWeimin Xu A Novel Data Encryption in HDFS. IEEE International Conference on Green Computing and Communicationsin 2013.

[10] Devaraj Das, Owen OeMalley, Sanjay Radia and KanZhang Adding Security to Apache Hadoop. in hortanworks

[11] Songchang Jin, Shuqiang Yang, Xiang Zhu, and Hong Yin Design of a Trusted File System Based on Hadoop .Springer-Verlag Berlin Heidelberg in 2013

[12] [AdvancedEncryption Standard,http://en.wikipedia.org/wiki/Advanced Encryption Standard

[13] Sharma Y. ; Kumar S. and Pai R.M; FormalVerification of OAuth 2.0 Using Alloy Framework .International Conference on Communication Systems and Network Technologies in 2011

[14] Ke Liu and Beijing Univ OAuth BasedAuthentication and Authorization in Open Telco API .IEEE International Conference on Communication Systems and Network Technologies in 2012

[15] Big Data Security: The Evolution of Hadoops Security ModelPosted by Kevin T. Smith on Aug 14, 2013

[16] Securing Big Data Hadoop: A Review of Security Issues, Threats and Solution by Priya P. Sharma and Chandrakant P. Navdeti in 2014

[17] https://console.developers.google.com

[18] https://developers.facebook.com/apps