Performance Analysis of the Encryption Algorithms as Solution to Cloud Database Security

Murtala Aminu Baba
Department of Computer and Communication Engineering, Faculty of Engineering and Engineering Technology, Abubakar Tafawa Blewa University, P.M.B 0248, Bauchi, Nigeria.

Abdulrahman Yusuf
Faculty of Science, Department of Computer Science, Yobe State University, P.M.B 1144, Damaturu, Yobe State Nigeria.

Aminu Ahmad
Department of Management and Information Technology, Abubakar Tafawa Balewa University Bauchi, Nigeria

Ladan Maijama'a
Department of Electrical/Electronic Engineering, Federal Polytechnic Bauchi, Nigeria

ABSTRACT
As the amount of data and information received and sent electronically increases in various Small and Medium Enterprises, companies and organizations, the adoption of cloud Database as a Service is becoming popular in the global business community. The operational burden encountered by users of Traditional Database Systems (TDS), such as configuration, provisioning, performance tuning, scalability, privacy and security, backup and access control, are transferred by Database as a Service to the service provider/operator. This offers lower hardware and software costs, remote online access to databases and reliable applications. Despite all the benefits of Database as a Service, security and privacy issues of the cloud database can present significant challenges. With the increase of unauthorized access to confidential data in the cloud, this paper presents encryption techniques that provide strong security against attacks using Transparent Data Encryption (TDE). To protect the confidentiality of the cloud database, TDE is used to transparently encrypt and protect data at rest, on hard disk, in transit and on backup media. TDE is effective, easy and provides high security levels to columns, tables and table space for data that require protection. This paper studies the performance of encryption algorithms (AES128, AES192, AES256 and 3DES168) with regard to CPU time, execution time and elapsed time. AES128 is found to have a better performance than other encryption algorithms.

Keywords: Encryption, TDE, TDS, AES, 3DES, Oracle12c

1. INTRODUCTION
Cloud computing (CC) as defined by National Institute of Standard and Technology (NIST) [10] is a model system that enables on-demand network access and convenient access of configurable computing resources such as storage, networks, servers, services and applications, which are released with minimum management and can be provided rapidly by service provider interaction. Benefits of CC according to [3] include reduction of system complexity for example hardware and networking equipment, economic and financial benefits, scalability, effective allocation of internal resources, better quality of service and flexibility.

The deployment methods of CC include Private cloud, Public cloud and Hybrid cloud. The seven cloud service deliveries namely, Platform as a Service, Infrastructure as a Service, Communication as a Service, Monitoring as a service, Software as a Service, Security as a Service and Database as a Service, [8].

The field of Software as a Service (SaaS) lays the new emerging field of Database as a Service (DaaS) and deliver the same and better functionalities as Database Management System (DBMS). DaaS market is swiftly increasing and the interest of many vendors have been attracted by changing the traditional data management client-server architecture, which involves data owner that manages and handles the user queries and DBMS respectively, to web based architecture where data owners outsource their data to third party service providers that manages and take control of the data, [6].

According to [2] despite all the benefits and advantages of moving database to the cloud in terms of better services, cost reduction, remote online access etc, yet they are not efficient to promote the adoption of DaaS. Security is the most serious issue to the global use of DaaS as data can be compromise with the service provider. Confidentiality of data is the fundamental aspect in cloud security as most data in the cloud are unprotected to unauthorized disclosure. One possible solution to loss of confidentiality in cloud database is by applying encryption method to the data from the client system before uploading to the cloud.

Although, encryption technique presents some challenges such as difficulty in performing over an encrypted data, key management, data availability and performance influence [16] and [17]. The research provides possible solutions to database security and alternative ways of querying encrypted data, secured key management and comparing the performance of encryption algorithms AES (128, 192, and 256) and 3DES168 in terms of their ability to protect and secure cloud data against any attack, efficiency, speed, power consumption and in doing so.

Moreover, in order to perform detail analysis, the research aims to study the relational cloud DaaS, services it offers and investigate the security issues associated with this service. The objectives of this paper include:

i. To investigate the security issues and challenges in relational cloud DaaS.
ii. To test, analyse and evaluate the performance of different encryption algorithms used in securing and protecting cloud database.
iii. To determine, provide and propose the most appropriate security measures relating to the threats in DaaS.

2. LITERATURE REVIEW

According to [9] cloud database are managed and designed by cloud providers such as Oracle, Microsoft, IBM, Amazon and Google. To host database resources and applications in the cloud, cloud environment provides resources and privileges to cloud database users. Other benefits of cloud database includes its reliable functionality, lower hardware and software costs, remote use of cloud database using internet connectivity and reduced staff management costs.

2.1 Security issues and challenges of cloud database

[15] Observes that the adoption and outsourcing of data to the cloud by the various companies, organization, banks, airlines, schools, hospitals and stock exchange are reluctantly hesitated due to the security threats

posed, which might result by attackers and or the cloud providers weak security measures. There are various forms of security issues in cloud database such as denial of service, repudiation, loss of integrity, confidentiality etc. It is however argued that the main security threat to the cloud database is the loss of confidentiality of data [16].

In addition to security of data in the cloud, database attack by hackers (passive or active) and serious limitations summed up the challenges that need to be addressed when outsourcing data to the cloud. For example, loss of data in the cloud, inability to access data in the event of failure and bankruptcy of the cloud service provider could attribute the hindrance of adopting cloud database.

2.2 Solutions to cloud database security

For effective protection of data confidentiality in the process of storage, transmission and processing in the cloud, encryption technique is the most adoptable method for securing sensitive data [24].

2.2.1 Encryption

The process of transforming an information or data into an unreadable text using encryption and decryption key, which cannot be easily understood, by the unintended reader or intruder is called the encryption, [7].

2.2.2 Why data encryption

The need to protect data theft, data fraud, data loss and data confidentiality are the main reasons for applying encryption methods to cloud data *[17]*.

2.3 Types of encryption

2.3.1 Asymmetric encryption: Asymmetric encryption involves the use of two different keys for securing data from cryptanalysis. These keys include the public key for other users to encrypt and the private key for the owner to decrypt it, [16]. Figure 1 shows an example of asymmetric encryption algorithm.

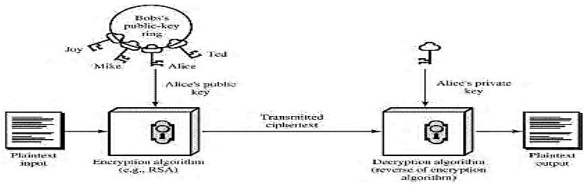

Figure 1: Asymmetric Encryption system [16]

*2.3.2 Symmetric encryption***:** According to [16] symmetric encryption provides a global technique for data transmission and storage and involves the process of secret communication using single-key encryption. The five ingredients of symmetric encryption are: plaintext, encryption algorithm, secret key cipher text and decryption algorithm. Figure 2 shows an example of symmetric encryption algorithm.

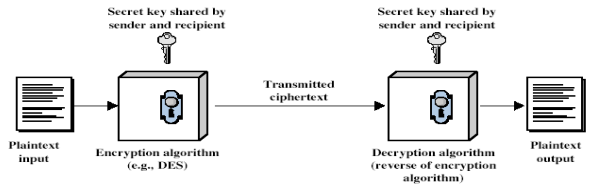

Figure 2: Simplified Model of Symmetric Encryption [16]

2.4 Challenges of data encryption

Despite the benefits and applications of encryption method in safe guarding cloud database, various reasons have hindered the deployment and sending data to the cloud. These factors include, performance impacts, difficulty in performing query over the encrypted data, key management issue, requirements of changes in data application, and data availability. [17].

2.5 Solutions to the challenges of data encryption:

Solutions to these challenges are provided based on the priority to the cloud data.

2.5.1 Query performance Execution of query over an encrypted data have been provided with various proposed techniques such as Aggregation queries- partial and fully homomorphic encryption, Range queries, trusted computing, cipher base and secure server [15], [1] and [4].

A profound flexible and reliable security solution to an encrypted data in the cloud according to [16] is the database should be worked in its encrypted form in the cloud without decrypting in the cloud. This approach would disallow the service provider to know the contents and query processing.

2.5.2 Key management: According to (RSA Security) key management issue in database encryption can be solved using external Hardware Security Module or Wallet. The function of Wallet is to store, manage and protect the

encryption keys and master keys. Thus, only authorised users are permitted and allowed to have access to encryption/decryption keys. Minimal access of encryption/decryption keys are preserved by Wallet.

2.5.3 Performance impacts: Transmission of additional bits of data to the cloud by the encrypted data has caused increase in data processing, memory capacity, power consumption and also increases computation delay of an encrypted data, [14]. [5] shows an example using Oracle 12c, Transparent Data Encryption which ensures data authenticity and confidentiality by the encryption method. However, this method has a negative impact on the size of encrypted database by adding 20 bytes Mandatory Access Control to each encrypted attribute in the database.

2.6 Test Plan

In order to provide a standard technique for protecting data in the cloud, encryption mechanism is studied to be the reliable approach to secure cloud databases. To analyse the level of security protection by the encryption method, a test plan is scheduled to explore the issues of employing database encryption and also to measure, compare and contrasts the performance in applying encryption scheme in terms of application performance and data security.

The test will also provide a set of recommendations that will guide and help organizations to develop strategies that will meet the needs of their customers as well safe guarding their sensitive data in the cloud. Oracle 12c will be used to test the performance of an encrypted data and unencrypted data in the cloud.

3. RESEARCH METHODOLOGY

The design analysis and performance for both the encrypted and unencrypted data are done with software that supports the database encryption. According to [18] different software can be used to approach encryption at various levels such as column, table and table-space. Microsoft SQL Server (2008-2012) and Oracle are the known technology employed in Transparent Data Encryption (TDE) for efficient encrypting of database content

3.1 Oracle 12c

It is argued that Oracle 12c is the first database designed for the cloud aimed at transforming business technologies and improving the effectiveness of the overall operational agility in cloud database [12]. It also provides new high-speed multitenant architecture, reliable, scalable and secured database platform. Benefits of Oracle 12c include the provision of high security levels in tables and columns, high quality storage and management, time saving and consolidated database management.

3.2 Transparent Data Encryption in Oracle 12c

TDE is used to encrypt and decrypt data on log files, backup media and disk. The benefits of TDE include the provision of protection against unauthorized access to sensitive information and the user is sure that the sensitive information is safe in case of unauthorized access to data [11].

The three important uses of TDE include authentication, validation and data protection. Data protection is the most widely used application of transparent encryption. Protection of valuable information in every organization is the key aspect to its productivity. Thus, TDE provides protection to columns, table and table spaces by transparently applying encryption technique.

3.3 Types of Transparent Data Encryption (TDE)

3.3.1 TDE Table-Space encryption: TDE also allows user to encrypt the entire tablespace as well as the entire objects created in the tablespace. It is useful to apply TDE tablespace encryption when sensitive data in tables need all to be encrypted. This process reduces tedious granular analysis to encrypt separate columns in a tablespace, thus applies the encryption to entire table in a reduced time and enhanced performance [13].

3.3.2 TDE column encryption: TDE column encryption is used to safeguard the confidentiality of data stored in a database such as social security numbers, credit card numbers, PIN, payroll information and health records stored in the table columns.

Task	SQL Command
Wallet setup in sql.ora	ENCRYPTION_WALLET_LOCATION = (SOURCE= (METHOD = FILE) (METHOD_DATA = (DIRECTORY = C:\APP)))
Initializing database Master key	ALTER SYSTEM SET ENCRYPTION KEY IDENTIFIED BY “AMINU9090”;
To *OPEN* wallet	ALTER SYSTEM SET ENCRYPTION WALLET OPEN IEDNITFIED BY AMINU9090;
To *CLOSE*	ALTER SYSTEM SET ENCRYPTION WALLET CLOSED IEDNITFIED BY AMINU9090;
Add encrypted column to existing table	ALTER TABLE table_name ADD (column_name datatype ENCRYPT
Encrypt unencrypted existing table	ALTER TABLE table_name MODIFY (column_name ENCRYPT)
Create Table and encrypt column	CREATE TABLE <table_name> (column_name datatype ENCRYPT)
Creating table from an existing table	CREATE TABLE <table_name> AS SELECT * <existing_table > ;

Table 1: SQL command quick reference in TDE

The SQL command used in the research work is shown in table 1. Thus, the SQL were used to query the data in the database.

4. DISCUSSION OF FINDINGS

Study has shown that the encryption algorithms have different performance in terms of efficiency, speed and execution time. Reference to the algorithm key sizes however contributed to the change in the encrypted table in terms of data size, elapsed time and CPU performance. From table 2, it can be seen that multiple readings of the original table size before encryption is taken to compare the performance with the encrypted columns.

TABLE SIZE AND EXECUTION TIME BEFORE ENCRYPTION			
ORIGINAL SIZE (Bytes)	EXECUTION TIME (µs)	CPU (seconds)	ELAPSED TIME (seconds)
665060	28201	0.03	0.03
665060	26798	0.01	0.03
665060	27071	0.07	0.03
665060	25288	0.01	0.03
665060	23334	0.07	0.03
Average			
665,060	**2,6138.4**	**0.038**	**0.03**

Table 2: Unencrypted Table

Table 2 shows relatively low CPU time, Execution time and the elapsed time. The data size is less than 1 Mega Byte and therefore can be easily transmitted with low performance constraints.

	AES 128				AES 192				AES256				3DES168			
Column No.	Table Size(Bytes)	CPU (s)	Elapsed (s)	Time(µs)	Size(Bytes)	CPU (s)	Elapsed (s)	Time(µs)	Size(Bytes)	CPU (s)	Elapsed (s)	Time(µs)	Size(Bytes)	CPU (s)	Elapsed (s)	Time(µs)
Encrypted Column 1	665060	0.34	0.45	50910	1625877	0.46	0.92	65279	1729090	0.49	0.67	66077	1732962	0.81	0.81	2221337
	665060	0.31	0.45	43739	1625877	0.31	0.40	45046	1729090	0.39	0.42	47830	1732962	0.53	0.63	30140
	665060	0.39	0.38	42330	1625877	0.31	0.47	52113	1729090	0.42	0.46	45782	1732962	0.67	0.69	37094
	665060	0.40	0.40	41192	1625877	0.40	0.38	42237	1729090	0.49	0.67	66077	1732962	0.84	0.93	70119
	665060	0.40	0.53	65808	1625877	0.40	0.37	39385	1729090	0.45	0.45	52242	1732962	0.74	0.73	41683
Average		**0.368**	**0.442**	**48795.8**		**0.376**	**0.508**	**48812**		**0.448**	**0.534**	**55601.6**		**0.718**	**0.768**	**480074.6**

Table 3: First Reading of encrypted Salary column

	AES 128	**AES 192**	**AES 256**	**3DES 168**
Time (secs)	0.442	0.508	0.534	0.768
Size (bytes)	0.665	1.626	1.729	1.733
CPU	0.368	0.376	0.448	0.718

Table 4: Extracted Time, Size and CPU performance

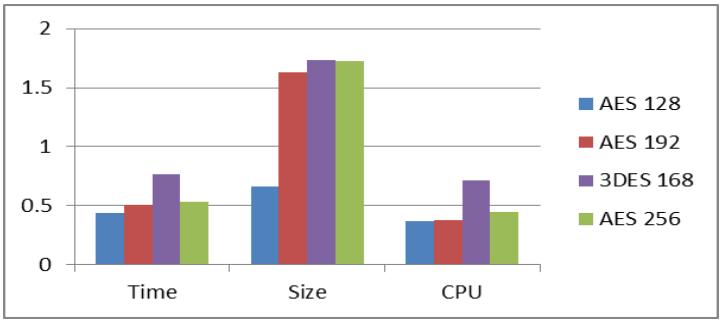

Figure 3: performance result of each encryption algorithm for Salary column

	AES 128				AES 192				AES256				3DES168			
Column No.	Table Size(Bytes)	CPU	Elapsed	Time (µs)	Size (Bytes)	CPU	Elapsed	Time (µs)	Size (Bytes)	CPU	Elapsed	Time (µs)	Size (Bytes)	CPU	Elapsed	Time (µs)
Encrypted Column 1 & 2	665060	0.62	0.65	38036	2292760	0.81	0.69	39155	2028020	0.70	0.63	32999	1840780	1.15	1.37	37079
	665060	0.68	0.75	44361	2292760	0.73	0.69	37327	2028020	0.65	0.59	32457	1840780	1.26	1.26	35782
	665060	0.59	0.57	30030	2292760	0.59	0.62	37076	2028020	0.65	0.59	32750	1840780	1.12	1.12	34891
	665060	0.65	0.63	34688	2292760	0.79	0.79	52722	2028020	0.81	0.60	32764	1840780	1.56	1.56	43496
	665060	0.60	0.57	29761	2292760	0.62	0.61	34399	2028020	0.63	0.59	29113	1840780	1.57	1.51	39375
Average		**0.628**	**0.637**	**35,375.2**		**0.708**	**0.680**	**40135.8**		**0.688**	**0.60**	**32,016.6**		**1.332**	**1.364**	**38124.6**

Table 5: Second Reading of Encrypted Salary and Job columns

	AES 128	**AES 192**	**AES 256**	**3DES 168**
Time	0.637	0.680	0.600	1.364
Size	0.665	2.293	2.028	1.841
CPU	0.628	0.708	0.668	1.332

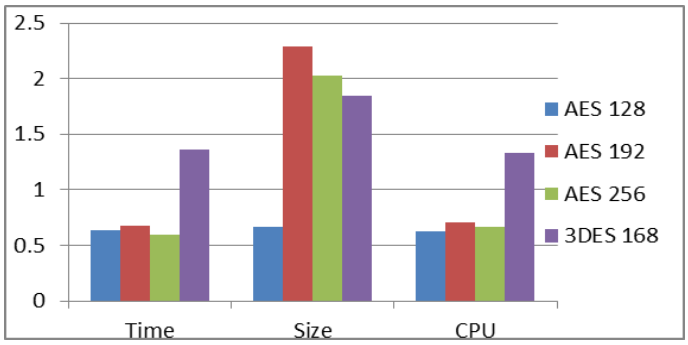

Figure 4: performance result of each encryption algorithm for Salary and Job columns

Table 6: Extracted Time, Size and CPU performance

	AES 128				AES 192				AES256				3DES168			
Column No.	Table Size(Bytes)	CPU	Elapsed	Time(µs)	Size(Bytes)	CPU	Elapsed	Time(µs)	Size(Bytes)	CPU	Elapsed	Time(µs)	Size(Bytes)	CPU	Elapsed	Time(µs)
Encrypted Column 1, 2 & 3	665060	0.73	0.80	65280	2813076	1.06	1.08	51830	2466261	0.81	1.20	37355	2628234	2.30	2.56	77570
	665060	0.76	0.80	67032	2813076	0.87	0.80	38008	2466261	0.71	0.83	32321	2628234	1.74	1.62	55537
	665060	0.82	0.80	65497	2813076	0.92	0.93	46968	2466261	0.85	0.83	32329	2628234	1.45	1.56	54506
	665060	0.67	0.80	75799	2813076	1.37	1.29	61289	2466261	0.74	0.83	33613	2628234	1.73	1.68	56169
	665060	0.79	0.81	67921	2813076	1.01	1.06	50993	2466261	0.74	0.83	34632	2628234	1.80	1.90	59099
Average		**0.754**	**0.802**	**68305.8**		**1.046**	**1.032**	**49817.6**		**0.77**	**0.904**	**34050**		**1.804**	**1.864**	**302,881**

Table 7: Third Reading of Encrypted Salary, Job and EmpNo columns

	AES 128	AES 192	AES 256	3DES 168
Time	0.802	1.032	0.904	1.864
Size	0.700	2.800	2.500	2.600
CPU	0.754	1.046	0.77	1.804

Table 8: Extracted Time, Size and CPU performance

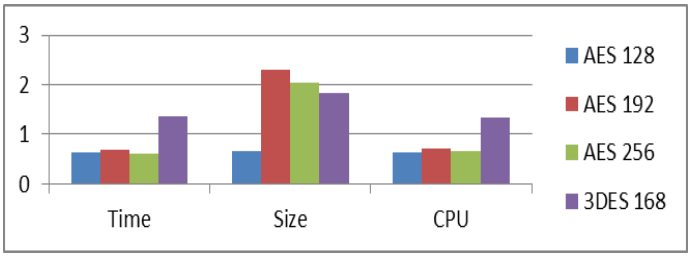

Figure 5: performance result of each encryption algorithm for Salary, Job and EmpNo columns.

	AES 128				AES 192				AES256				3DES168			
Column No.	Table Size(Bytes)	CPU	Elapsed	Time(µs)	Size(Bytes)	CPU	Elapsed	Time(µs)	Size(Bytes)	CPU	Elapsed	Time(µs)	Size(Bytes)	CPU	Elapsed	Time(µs)
1st Reading	665060	0.368	0.442	48795.8	1625877	0.376	0.508	48812	1729090	0.448	0.534	55601.6	1732962	0.718	0.768	480074.6
2ndReading	665060	0.628	0.637	35,375.2	2292760	0.708	0.680	40135.8	2028020	0.688	0.60	32,016.6	1840780	1.332	1.364	38124.6
3rd Reading	665060	0.754	0.802	68305.8	2813076	1.046	1.032	49817.6	2466261	0.77	0.904	34050	2628234	1.804	1.864	302,881

Table 9: Summary of the Average readings for the Encrypted column

5. EVALUATION

This section discusses the results obtained from the repeated tests taken on the encrypted columns of the various algorithms used in the analysis. Consequently, the results presented in the previous sections show the impact of column encryption on execution and elapsed time, increased table sizes and CPU performance.

The first reading of the experiment was carried out to measure performance of the unencrypted table, which is shown in table 2. However, some parameters have changed when the `Sal` column was encrypted. Table size has significantly shown no change using AES128 algorithm although time elapsed and CPU time has drastically increased by more than 10%. The result shows that 3DES168 consumes more resources (elapsed time and CPU) due to its highest level of encrypted data size. Therefore, it can be concluded here that, 3DES168 showed poor performance results since it consumes more power than AES 128,192 and 256. AES 128 has a better performance since it uses less time and CPU. [22] have expressed a similar view on 3DES168 which consumes more processing time and CPU, thus makes execution process very slow.

Figure 3 presented the result of the average second experiment. The output clearly indicates that 3DES168 has the highest level of elapsed and CPU processing time over its counterpart AES. Time consumed by 3DES168 is twice the amount of AES128. In terms of increased database size, AES 192 is the greatest with the value of 2.3MB while AES 192 has 0.667MB.

Given this evidence, it can be seen that the table 2 have the same size before encryption while after been encrypted with different database encryption algorithms, their sizes and performances have changed. It is clear therefore be recognised that AES128 has better performance than other encryption techniques. [19] concluded after studying encryption algorithms that 3DES168 is the least in terms of efficiency and performance throughput while AES has better performance efficiency. Although, they were analysed based on the selected encryption algorithms 3DES, AES, DES and Blowfish in which separate analysis on AES128, 192 and 256 were not conducted. Similarly, research by [23] concluded after conducting throughput analysis of various encryption algorithms using windows 7 Home Premium (32bit), 2.20GHz C.P.U., 4GB RAM Core-2-Dou Processor, that Blowfish has better performance than other algorithms and AES is followed with 3DES168 as the least efficient of all the algorithms studied. Their implementation results were tested and the encryption algorithm performances were optimized to have maximum efficient results. The work of [23] indicated that Blowfish is the best encryption algorithm based on their study however, Blowfish was designed in 1993 which is quite long after which NIST recommended AES in 1997 as the best encryption algorithm in the 21st century. Thus, this research has strong evidence that 3DES168 has the least performance efficiency and AES has better performance.

[21] recommended that the use 3DES168 in protecting sensitive information against attack should be minimal as the research found that 3DES is the weakest and slowest than the other encryption algorithms. Additionally, the author has drawn attention to the fact that the security performance of 3DES is weak and is easier to break.

The evidence shown in the overall result in figure 5 indicated that execution and CPU time of AES is lower than 3DES168 and therefore it is inferred that the performance of AES is better than 3DES168. Turning to AES algorithms, in this research, AES128 is studied to have the highest quality of encryption performance and strongest security protection.

In a study by [20] identifies the gap by releasing a white paper, which reveals that encrypting databases of the healthcare organizations introduces an additional impact performance and computational load. The study also found that enabling TDE on the computer system increases CPU overhead by 800%. Moreover, the evidence seems to indicate that encrypting database ends in a longer general reporting job runtime, high CPU utilization, high computation overhead and high impacts on the overall performance.

The research propose a standard solution to the limitations of using TDE in encrypting data in a database. According to the findings, an alternative solution to TDE additional performance is by using Intel Advance Encryption Standard New Instructions (Intel® AES-NI), thus reduces the impact on CPU performance by more than 50%. (Intel® AES-NI) delivers affordable data protection, security and fast CPU performance. (Intel® AES-NI) is activated in the database through database configuration updates and BIOS. The use of (Intel® AES-NI) when compared to software-only encryption requires less than half as much as CPU usage. Additional computing overhead can be off-loaded easily when (Intel® AES-NI) hardware accelerated encryption/decryption is activated which also reduces server build-out costs and considerably shortens the reporting job duration.

Thus, it could be concluded that, a solution to cloud database security threat is encryption technique. Although encrypting data at rest or in-transit causes computational and affects the overall performance of the system, it is found that activating (Intel® AES-NI) reduces the performance impact of encryption increases the CPU performance by more than

6. CONCLUSION

The benefits of cloud database as a service such as reduced staff management costs, hardware and software costs, 24/7 remote access to cloud database and lower cloud license costs have transferred most of the traditional database system to the reliable cloud database. Overcoming cloud limitations and loss of confidentiality with encryption methods would strengthen the effective use and adoption of cloud database as a service.

Numerous mechanisms are available for securing cloud database as a service such as symmetric and asymmetric key cryptography. Out of the various encryption algorithms present, this paper has conducted a comparison between AES and 3DES168. From the analysis and the literature reviewed, it can be concluded here that AES128 has better performance than AES192, AES256 or 3DES168. The solutions to data encryption problems such as key management, querying encrypted data and performance impacts have also been stated. To ensure the security of cloud database, AES can be used sufficiently to provide maximum security and better CPU, execution and elapsed time using Transparent Data Encryption. In the future, we propose to further the research on the adoption of AES and 3DES on larger database system such as organization with huge amount of data entries

7. REFERENCES

[1] Agrawal, D., Abbadi, A. E., and Wang, S. (2013) Secure and privacy-preserving database services in the cloud, *2013 IEEE 29th International Conference on Data Engineering (ICDE)* [online]. pp.1268-1271

[2] Arasu, A., Eguro, K., Kaushik, R., and Ramamurthy, R. (2013) Querying encrypted data, *IEEE 29th International Conference on Data Engineering (ICDE)* [online]. pp.1262-1263

[3] Ahmad, T., Haque, M. A., Al-Nafjan, K., and Ansari, A. A. (2013) Development of cloud computing and security issues, *information and knowledge management* [online]. **3**(1), pp.34-42

[4] Arasu, A., Blanas, S., Eguro, K., Kaushik, R., Kossmann, D., Ramamurthy, R., and Venkatesan, R. (2013) Orthogonal security with cipherbase, *6th Biennial Conference on Innovative Data Systems Research* (*CIDR* '13).

[5] Bouganim, L., and Guo, Y. (2009) Database encryption. *Encyclopedia of cryptography and security* [online]. pp.1-9

[6] Ferrari, E. (2009) Database as a Service: Challenges and solutions for privacy and security. *Asia-Pacific Services Computing Conference, 2009. APSCC 2009. IEEE* [online]. pp.46-51.

[7] Goodrich, M.T. & Tamassia, R. (2011) *Introduction to computer security*, Pearson, Boston, Mass

[8] Kuyoro, S. O., Ibikunle, F. and Awodele, O. (2011) Cloud computing security issues and challenges. *International Journal of Computer Networks* [Online]. **3**(5), pp. 247-253

[9] Mateljan, V., Cisic, D., and Ogrizovic, D. (2010) Cloud Database as a Service (DaaS)-ROI. *Proceedings of the 33rd International Convention in MIPRO*, IEEE [Online]. pp.1185-1188

[10] NIST (2013) NIST Cloud Computing Program [online]. [Accessed 09th July 2013]. Available at: <http://www.nist.gov/itl/cloud/index.cfm>

[11] Oracle (2013a) Oracle Advance Security [online]. [Accessed 09th July 2013]. Available at: < http://www.oracle.com/technetwork/database/options/advanced-security/index.html >

[12] Oracle (2013b) Oracle Press Release [online]. [Accessed 15th August 2013]. Available at: <http://www.oracle.com/us/corporate/press/1967380>

[13] Oracle (2013c) Database Advanced Security Administrator's Guide <http://docs.oracle.com/cd/E11882_01/network.112/e10746/asotrans.htm#autoId3>

[14] Pathan, A. S. K., Lee, H. W., & Hong, C. S. (2006) Security in wireless sensor networks: issues and challenges [online]. *The 8th International Conference proceedings in Advanced Communication Technology, 2006. ICACT 2006.* **2**, pp.1043-1048.

[15] Rehman, A. U., and Hussain, M. (2011) Efficient Cloud Data Confidentiality for DaaS. *International Journal of Advanced Science and Technology* [online]. **35**, pp.1-10

[16] Stallings, W., Brown, L.V., Bauer, M. and Howard, M. (2008) *Computer security: principles and practice*, Upper Saddle River, N.J.: Pearson Prentice Hall

[17] The Sans Institute (2009) Transparent Data Encryption [online]. pp.1-33. Available at: <https://www.sans.org/webcasts/92778.pdf>

[18] Developer Network (2014) Transparent Data Encryption [online]. [Accessed 09th July 2014]. Available at:<http://msdn.microsoft.com/en-us/library/bb934049.aspx>

[19] Paliwal, S. and Gupta, R. (2013) A review of some popular encryption techniques. International Journal of Advanced Research in Computer Science and Software Engineering. [online]. **3**(2), pp.147-149

[20] Intel® Corporation (2009) Encrypt Healthcare Data with performance using Intel and Xeon Processors, white Paper, USA.

[21] Carl D. (2011) Oracle 11g Transparent Data Encryption, UK Oracle User Group (UKOUG).

[22] Singh S.P and Maini R. (2011) Comparison of data encryption algorithms. *International journal of computer science and communication.* **2**(1), pp125-127.

[23] Singh G., Kumar A. and Sandha K.S (2011) A study of new trends in Blowfish Algorithm. *International Journal of Engineering and Research and Applications* (IJERA). **1**(2), pp.321-326.

[24] Gharehchopogh F.S and Bahari, M. (2013) Evaluation of the data security methods in cloud computing Environments. *International Journal in Foundations of Computer Science and Technology (*IJFCST). **3**(2), pp.41-51