Performance Evaluation of Encryption Techniques for Confidentiality of Very Large Databases

Malik Sikander Hayat Khiyal, *Member, IACSIT*, Aihab Khan, and Khansa Shabbir

***Abstract*—Many companies rely heavily on the functions of databases, hence database management and maintenance has become a vital component of their business models. There are many architectures, techniques, and tools available to ensure security. But to select the right solution for the right application is the most important issue of the time. This research is conducted with the objective to provide with the best encryption algorithms for confidentiality of very large databases. When different techniques were analyzed in terms of response time, memory usage and efficiency, 3DES (triple data encryption standard) performed better than advanced encryption standard (AES) and Blowfish. AES provides the highest security but to use it the organization would have to compromise on response time and resource usage. If the organization is short of resources and cannot spend much time in encrypting data though the confidentiality is not much required then 3DES provides the best solution. Blowfish holds the in between place. If neither the security is a maximum risk, nor the resources are a problem, then Blowfish should be preferred.**

***Index Terms*—Databases, confidentiality, AES, 3DES, blowfish.**

I. Introduction

Transaction logging, warehousing and processing of information about these transactions is the lifeline of corporate strategy of many business companies and hence plays a vital role in their success. Essential records detailing a company's product inventory, its user history, configuration settings, supplier information, shipment tracking, or any other useful collections of information are most often retrieved from and stored in the databases. Databases provide the facility of storing enormous amount of information. Hence allowing the information to be stored, viewed, searched and manipulated according to the needs and goals of the business. Many companies depend upon the functions of the databases so heavily that in the absence of these databases their daily business operations cannot be executed. Thus database management and maintenance have become an imperative component of the business models. For most of the business as well as the home computer users, data security is one of the critical issues. If the information such as payment information, bank account details, client information or even personal files falls into the wrong hands than it would be difficult and potentially risky to restore this information. Data lost due to natural disasters such as fire or flood is upsetting, but losing it to malware infection or even to hackers can have even much more greater consequences.

In this age of information technology everyone wants perfection of time and effort. Security of digital information thwarts many problems and so are devised the solutions. A lot of work is done for the security of different digital mediums but archives security still poses many problems. To select the right solution for the right application is the most important issue of the time. So the performance of available encryption techniques fulfilling the confidentiality of large databases has been analyzed in this research work. The results will be helpful for large organizations maintaining the security of their databases.

For the research area archives have been selected. Most of the organizations today store their data, both confidential and non-confidential, in the form of digital archives. As a consequence security of archives is an important issue requiring special attention. More importantly, availability of data to the right person within shortest time must be made sure. The archival records are useful for research and historical purposes in addition to the monetary, legal and administrative purposes for which these records were originally created and were used. Archives basically provide a key with which the past as well as the present events can be examined. A variety of researchers have taken the advantage of the archival sources in addition to its administrative uses. These researchers may include students at all levels, biographers, scholars, local historians, documentary filmmakers, independent writers and genealogists. Since the archival documents can be used for many purposes by a diverse audience, the records of individual's personal paper as well as the record of the organization that lacks in having their own institutional archives are often actively required by these archival programs. These types of institutions focus on collecting records that document a particular topic rather than documenting the activities of a parent organization.

II. Related Work

Wong [1] investigated practical solutions for integrating picture archiving and communication systems (PACS) and cryptographic techniques so that the security of medical images can be improved. In addition to this, the timing performance of encrypting, decrypting and transmitting cryptographic protocols over 81 volumetric PACS datasets has been measured also. Before the encryption process, lossless data compression has been applied also. The transmission performance has been measured against three different types of networks with different bandwidths. These are Ethernet, narrow-band integrated services digital network

Manuscript received July 27, 2011; revised November 2, 2011.

Authors are with the Department of Software Engineering in Fatima Jinnah Women University, The Mall, Old Presidency, Rawalpindi-Pakistan (e-mail: m.sikandarhayat@yahoo.com; aihabkhan@yahoo.com; khansa@gmail.com).

and OC-3c asynchronous transfer mode. An image dataset is retrieved from the UCSF PACS archive and subjected to lossless compression. The secret-key algorithm known as the international data encryption algorithm (IDEA): is used to encrypt the compressed images, where the secret key is randomly generated for each image encryption. The popular and robust public-key algorithm known as Rivest, Shamir, and Adleman (RSA) is used to encrypt the randomly generated secret key, *i*, which is a 1,024-bit number. This public-key-encrypted secret key is sent along with the ciphertext (encrypted image dataset) to the receiver. The receiver uses an individual private key to recover this encrypted secret key and then applies that key to run the fast secret-key algorithm to decrypt the large ciphertext. In this experiment, the sender and the receiver are located in separate workstations in the PACS networks used. This study takes a software system view of creating trust in digital radiology images and investigates the holistic approach of integrating a spectrum of cryptographic algorithms to meet the various security requirements of patient records. The performance of the cryptosystem is evaluated in encrypting several imaging modalities: 16-bit MRI, 8-bit post processed MRI, and PET, using cryptographic protocols operated within the hospital's integrated PACS environment. The confidentiality model proposed can be utilized for databases as well. The paper proposed a well applicable technique for securing digital images but it does not address the security problem of other digital assets like digital archives. Performance is evaluated on different imaging modalities but not on different encryption techniques.

Iyer *et al*. [2] argued that that adding privacy as an addendum results in suboptimal performance. They have proposed a storage model and have developed a suitable key management technique that minimizes the possibility of the data as well as key compromise. The main contribution of the paper is a new secure DBMS storage model that aids an efficient implementation. The approach involves grouping if sensitive data so that the number of necessary encryption operations are minimized and thus the cryptographic overhead can be lowered down. The authors have assumed a client-server scenario. The client has both the combination of sensitive as well as non-sensitive data that is stored in a database at the server having the sensitive data in encrypted form. Whether or not both the parties are co-located, it does not make any difference in terms of security. The additional responsibility of the server is the protection of the client's sensitive data i.e., the confidentiality of the client's data is ensured and unauthorized access is prevented. This is accomplished through combining the authentication, encryption and access control. The model suggests an effective approach for minimizing encryption cost. A new DBMS storage model (PPC) that aids efficient incorporation of encryption, has been proposed in this paper. The approach is based on grouping sensitive data so that the number of encryption operations can be minimized. Thus, the encryption overhead can be reduced greatly. Finally the proposed model is compared and contrasted experimentally with the previous models proposed. A number of important issues regarding query processing, access and storage are discussed. However, in this paper a model for security of databases is proposed but the performance of different encryption techniques is not analyzed. Small databases are accessed but large archives are not studied in this paper. The approach, if utilized, will change the size of the database while selectively encrypting the fields. This will affect the size parameter for performance measurement.

Adeel *et al*. [3] presents efforts for developing a mathematical search engine for mathematical content retrieval. In the paper a Math GO! system has been presented to search and present the mathematical information encoded in the mathematical expressions. To summarize the search engine architecture, the search engine is based on the text IR system principles with techniques for making it math aware. The search engine uses a hybrid model where user can move to and fro between a browsing and querying model. User can navigate to his required topic and can use the search facility provided. An eqn ID is given to every equation. The equation ID is stored in a separate table. Equations are identified with unique id's system wide. When the user inputs a new formula, an ID of 0 is assigned to it. Regular expression matching is used to extract the entered formula. Then the Query Processor component forms the query vector for the entered formula. Then the query vector is transformed to uniforms size. Convert the word frequency values to qf-idf and then normalize the qf-idf values. QF-IDF is employed for better weighting. Equations are represented with unique id's system wide. Equations are stored in a table. The approach defined uses the concept of template based math block identification, searching from mathematical topic based clusters, vector representation and relevance ranking. The search system interacts with the user through a simple query mechanism and thus provides a ranked list of the results. The system is comprised of a modular architecture so that the math results can be organized, queried, compare and present to the user. The issues that generally occur in math search engine are addressed in this approach. The performance of the search engine is also evaluated. It provides an archival approach. However, in this research work the confidentiality of the database is not encountered. Performance with respect to encryption techniques is not undertaken.

III. Proposed Framework

In this paper a framework is proposed for confidentiality of large databases by combining symmetric and asymmetric encryption techniques to join the advantages they provide with separately, for storage in archives.

Fig. 1 explains the registration process of object. The input object is encrypted using symmetric encryption scheme with AES, DES, and Blowfish. The performance of these encryption algorithms will be recorded according to different parameters. The results for encryption techniques are gathered for encryption as well as decryption and compared for time taken, disk memory utilized for storage, key size used and data object size taken as input. On the basis of these results conclusion is derived. The results will be stored in a separate database. Analyzing the performance results, the user selects an encryption technique either of the three. The object is then encrypted with the user's selected technique for confidentiality.

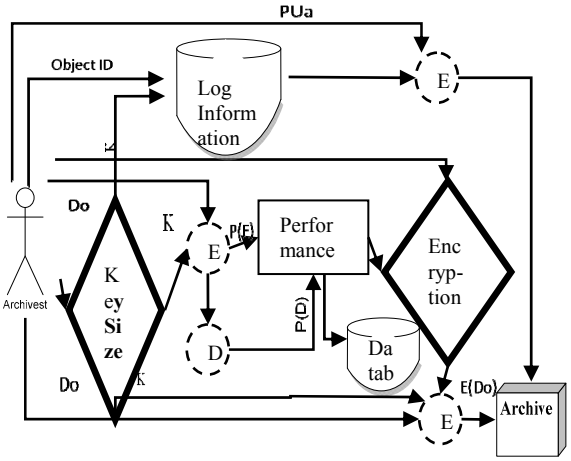

Fig. 1. Data object storage process.

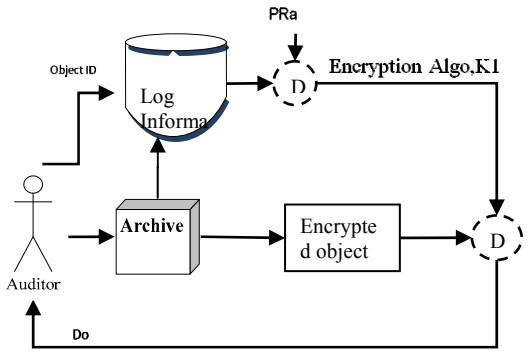

Fig. 2. Data object retrieval process.

A log will be maintained containing the necessary information about the data object, e.g., object id, type of encryption algorithm, key used for encryption, timestamp etc. This log will be encrypted using asymmetric encryption scheme RSA, to secure the information. The encrypted log and the object both will be stored on an archive.

Fig. 2 gives details about the object retrieval process. The auditor will ask for a specific object from the archive through object ID. If the user is authenticated to access the object the log information of the specific object will be decrypted using the private key of the user. Using the information about the encryption algorithm and key from the log information, that specific object will be decrypted. The user will be able to access the required object.

IV. Technique

The archivist will be able to store the data object on the archive.

- User specifies his user name and password for public key.
- User selects a data object.
- System encrypts/ decrypts the selected object.
- The performance results are shown.
- Performance results are saved in a database.
- User selects an encryption algorithm on the basis of performance, for storage.
- Log Information of the object is saved.
- Log Information is encrypted using user's public key.
- Both encrypted object and log information is saved on archive.

Fig. 3. Data object storage algorithm.

Fig. 3 clarifies the algorithm employed for storing a data object on the archive.

- User verifies his user name and password.
- User selects a data object.
- Log Information is decrypted with user's private key.
- Encrypted object's algorithm and key size is retrieved from log information.
- Object is decrypted.
- User accesses the object.

Fig. 4. Data object retrieval algorithm.

The procedure for retrieving an object from the archive is enlightened in Fig. 4.

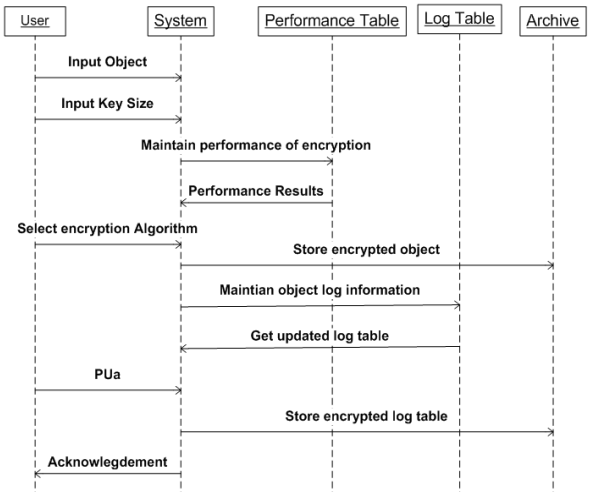

Fig. 5. Data object storage sequence diagram.

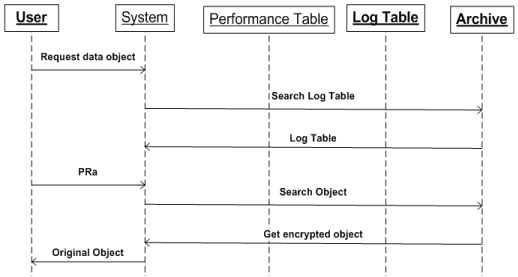

Fig. 6. Data object retrieval sequence diagram.

V. Results

In the research, ten objects with varying sizes ranging from 5Mb to 12Mb were selected. All three algorithms were tested on all objects for response time and resource usage. The results were conducted on a personal computer with 0.98GB RAM, Intel® Core™2 CPU 6420 @2.13GHz each.

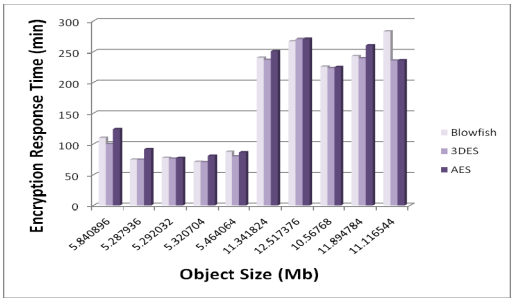

Fig. 7. Encryption response time vs. object size plot for each algorithm.

If a graph is drawn between response time and objects with respect to algorithms for encryption process, it could be seen that the algorithm 3DES shows less or equal to response than AES and Blowfish.

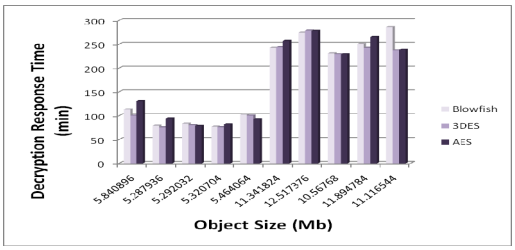

Fig. 8. Decryption response time vs. object size plot for each algorithm.

Similar results are observed for decryption process. 3DES again showing better response time as compared to AES and Blowfish.

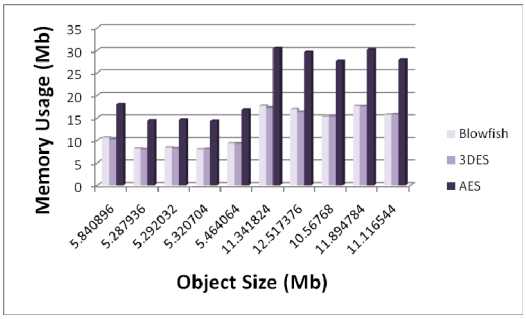

Fig. 9. Memory usage vs. object size plot for each algorithm.

If resource usage is plotted against objects with respect to all algorithms, it could be noticed that AES encryption consumes highest disk memory as compared to 3DES and Blowgish. 3DES shows a slighter less memory usage than Blowfish.

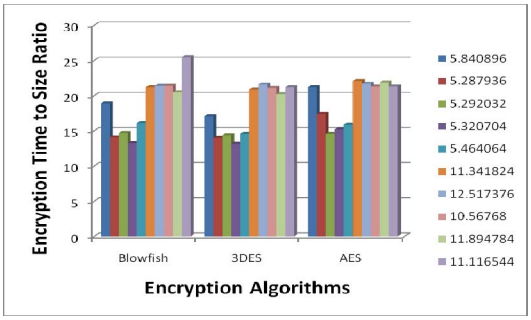

Fig. 10. Encryption response time to object size plot for each algorithm.

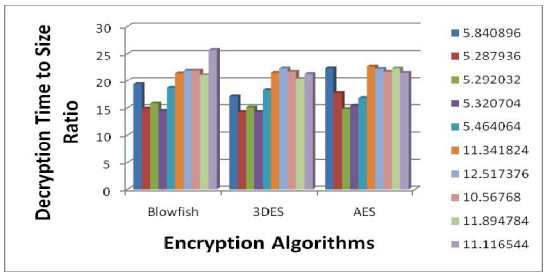

Fig. 11. Decryption response time to object size plot for each algorithm.

When time (encryption) to size ratio for each object is plotted, it becomes clear that 3DES shows a less variation on changing object size. So 3DES can be taken as more efficient as compared to AES and Blowfish. Next to 3DES, AES performs better than Blowfish. Blowfish shows the highest variation.

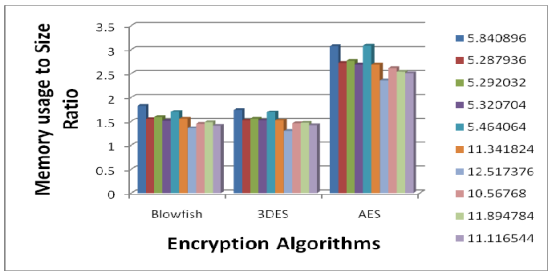

Fig. 12. Memory Usage to object size plot for each algorithm.

When time (decryption) to size ratio for each object is plotted, it becomes clear that 3DES shows a less variation on changing object size. So 3DES can be taken as more efficient as compared to AES and Blowfish. Next to 3DES, AES performs better than Blowfish. Blowfish shows the highest variation.

When memory (encryption) to size ratio for each object is plotted, it becomes clear that 3DES shows a less variation on changing object size. So 3DES can be taken as more efficient as compared to AES and Blowfish. Next to 3DES, Blowfish performs better than AES. AES shows the highest variation.

VI. Conclusion and Future Work

In this research work three well known algorithms, AES, 3DES and Blowfish, were analyzed for archival storage. When different data objects were encrypted using these algorithms, 3DES performed better in case of response time. AES was next to 3DES. Blowfish showed high response time. Similar results were observed for decryption response time.
For disk space consumption, AES utilized highest disk space for storing encrypted data. 3DES again showed less memory usage as compared to Blowfish.

For analyzing efficiency in terms of response time, time to size ratio was calculated. 3DES showed a relatively consistent behavior. AES showed somewhat varying behavior. Blowfish performed very inconsistently.
For analyzing efficiency in terms of memory usage, resource usage to size ratio was calculated. 3DES again showed a relatively consistent behavior. Blowfish showed somewhat varying behavior. AES performed very inconsistently.

As far as the security of the algorithms is considered, AES is considered most secure technique since attempts to break its key have been failed up till now. However, theoretically an inherent flaw in AES has been demonstrated by a team of researchers from France, Israel and Germany. But they have suggested that there is nothing to worry about this weakness of AES. AES still has flaws theoretically. The bottom line is that AES isn't broken (see Germain, [4]).

No successful cryptanalysis on the full-round version of Blowfish is known yet. Only in one publication of C code, a sign extension bug has been identified (see Schneier [5]).

Vaudenay [6] found a known-plaintext attack that requires 2^{S+1} known plaintexts to break, where the number of rounds is represented by r. He also found that with only 2^{S+1} known plates a class of weak keys can be detected and broken down with the same attack. Where this attack cannot be used against the regular Blowfish, as it assumes the knowledge of the key-dependent S-boxes. Rijmen [7] introduced a second-order differential attack that can break four rounds only in his Ph.D. thesis. Except the brute-force search there is

no recognized way to break the full 16 rounds (see [6]).

Generally a Triple DES consisting of three independent keys has a key length of 168 bits where each DES keys is of 56 bits. But the effective security provided is only 112 bits due to the meet-in-the-middle attack. If key 1 (K_1) and key 2 (K_2) are independent, and key 3 (K_3) is equal to K_1, the size of the key is reduced to 112 bits. However, this option is susceptible to certain known-plaintext or chosen-plaintext attacks (see Vaudenay [6]) and thus is selected by NIST to have only 80 bits of security (see Merkle and Hellman [8]).

About 2^{32} known plaintexts, 2^{20} single DES encryptions, 2^{113} steps and 2^{SS} memory(see Oorschot and Wiener [9]) are required for the finest attack if all three keys that are independent. This is impractical and NIST considers this keying option to be appropriate through 2030 (see Barker *et al*. [10]). There is a memory capable attack if the invader seeks to find out any one of many cryptographic keys. This will discover 2^{2S} keys, given a handful of chosen plaintexts per key and around 2^{84} encryption operations (see Biham [11]).

So, to conclude with AES provides the highest security but to use it the organization would have to comprise on response time and resource usage. If the organization is short of resources and cannot spend much time in encrypting data though the confidentiality is not much required then 3DES provides the best solution. Blowfish holds the in between place. If neither the security is a maximum risk, nor the resources are a problem, then Blowfish should be preferred.

This research was conducted on a limited number of encryption techniques. In future other algorithms available should also be analyzed for performance. It is possible that a technique outdated for other applications might provide best results for archival data storage. In addition, the medium selected was mainly databases. For further work other media like images etc. should also be utilized.

REFERENCES

[1] S. T. G. Wong, "A cryptologic based trust center for medical images," *Journal of the American Medical Informatics Association (JAMIA),* vol. 3, no. 6, pp. 410-42, 1996

[2] B. R. Iyer, S. Mehrotra, E. Mykletun, G. Tsudik, and Y. Wu, "A framework for efficient storage security in RDBMS," *Lecture Notes in Computer Science*, vol. 2992, pp. 627-628, 2004

[3] M. Adeel, H. S. Cheung, and M. S. H. Khiyal, "Math Go! prototype of a content based mathematical formula search engine," *Journal of Theoretical and Applied Information Technology*, vol. 4, no. 10, pp: 1002-1012, 2008

[4] J. M. Germain, (2009). Is AES encryption crackable? Article published in TechNewsWorld, November 5, 2009, 11:06 AM, http://www.betanews.com

[5] B. Schneier, From best-of-security-request@suburbia.net Mon July 8 01-58:54, 1996. http://www.schneier.com/blowfish-bug.txt

[6] S. Vaudenay, On the Weak Keys of Blowfish**.** Fast Software Encryption (FSE'96), LNCS 1039, D. Gollmann, Ed., Springer-Verlag, 1996, pp. 27—32, 1996

[7] V. Rijmen, Cryptanalysis and Design of Iterated Block Ciphers Ph.D thesis, ESAT/COSIC lab the Katholieke Universiteit Leuven (K.U.Leuven.), 1997

[8] R. Merkle and M. Hellman, On the Security of Multiple Encryption. Communications of the ACM, Vol 24, No 7, pp 465–467, July 1981.

[9] P. Van Oorschot and M. J. Wiener, A known-plaintext attack on two-key triple encryption, eurocrypt'90, LNCS 473, 1990, pp 318–325, 1990

[10] E. Barker, W. Barker, W. Burr, W. Polk, and M. Smid, Recommendation for Key Management — Part 1: General (Revised), , NIST Special Publication 800-57. March, 2007.

[11] E .Biham, How to Forge DES-Encrypted Messages in 2^{28} Steps, 1996. Technical Report CS 884, Department of Computer Science, Technion, Haifa, Israel, Aug., 1996

Dr. M. Sikandar H. Khiyal born at Khushab, Pakistan. He is Chairman Dept. Computer Sciences and Software Engineering in Fatima Jinnah Women University Pakistan. He Served in Pakistan Atomic Energy Commission for 25 years and involved in different research and development program of the PAEC. He developed software of underground flow and advanced fluid dynamic techniques. He was also involved at teaching in Computer Training Centre, PAEC and International Islamic University. His area of interest is Numerical Analysis of Algorithm, Theory of Automata and Theory of Computation. He has more than hundred research publications published in National and International Journals and Conference proceedings. He has supervised three PhD and more than one hundred and thirty research projects at graduate and postgraduate level. He is member of SIAM, ACM, Informing Science Institute, IACSIT. He is associate editor of IJCTE and Co editor of the journals JATIT and International Journal of Reviews in Computing. He is reviewer of the journals, IJCSIT, JIISIT, IJCEE and CEE of Elsevier. He can be contacted at m.sikandarhayat@yahoo.com, Fatima Jinnah Women University, Rawalpindi Pakistan.

Mr. Aihab Khan works in Department of Computer Sciences Fatima Jinnah Women University, Pakistan. His research interests are in the field of Data mining, Data Warehousing as well as Information Security. He can be contacted at aihabkhan@yahoo.com, Fatima Jinnah Women University, Rawalpindi Pakistan.

Khansa Shabbir is a software engineer graduate from Department of Software Engineering, Fatima Jinnah Women University, Pakistan. She can be contacted at salmakayani273@hotmail.com