Journal of
Data Acquisition and Processing

ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777497

DEVELOPING A FRAMEWORK FOR SECURING AND SEARCHING UNSTRUCTURED CLOUD DATA USING ADVANCED ENCRYPTION STANDARD

David Livingston J
Research Scholar, Department of Computer Science & Engineering, Karunya Institute of Technology & Sciences, Coimbatore, Tamilnadu, India. Scopus Author ID: 57210705383
ORCID: https://orcid.org/0000-0001-9074-2523

Kirubakaran E
Director, Grace College of Engineering, Thoothukudi, Tamilnadu, India.
Scopus Author ID: 35173015500

Immanuel Johnraja J
Professor & Head, Department of Computer Science & Engineering, Karunya Institute of Technology & Sciences, Coimbatore, Tamilnadu, India. Scopus Author ID: 46161242800
ORCID: https://orcid.org/0000-0002-8548-3333

Abstract— In general, databases are stored and managed on computers having any one of the three architectures: 1.Server Architecture, 2.Parallel Architecture, and 3.Distributed Architecture. The server architecture has two-tier - Tier 0 and Tier 1. In this architecture, clients act as the front-end at Tier 0 and the DBMS acts as a back-end at Tier 1. Parallel architecture allows multiple processors to run simultaneously, and the resources such as memory, processor and databases be shared among them. But in shared nothing parallel computers, the processors share neither memory nor storage device among them. Distributed architecture is the third type of architecture, which is similar to that of the Shared Nothing Architecture in parallel computers. Databases maintained by computers connected using distributed architecture are known as distributed databases. Distributed databases require a mechanism for maintaining the privacy of data in transit and data at rest. One of the mechanisms available for securing data privacy of data stored in a distributed environment is Encryption. In this paper, the authors proposed a framework in which Unstructured Data can be converted to Structured Data and stored in a database in an encrypted format using Advanced Encryption Algorithm - a kind of block cipher.

Keywords- Database Management System, Server, Parallel and Distributed Architecture, Data Privacy, Symmetric Encryption, Searchable Encryption

I. INTRODUCTION TO THE STUDY

Database Management System is a general purpose software used to store and retrieve a collection of interrelated data from a centralized repository called database. Each database managed by a DBMS can contain information relevant to an enterprise. For instance, a database that stores information about sales of an enterprise may contain data about products, customers and the purchase made by them. DBMS is one of the tools required for developing application software that stores and retrieves enterprise data. In a client/server environment, DBMS acts as a server (back-end) and the application program as a client (front-end). The role of DBMS in application development is to enable the programmer to store and retrieve large volume of data from a database efficiently.

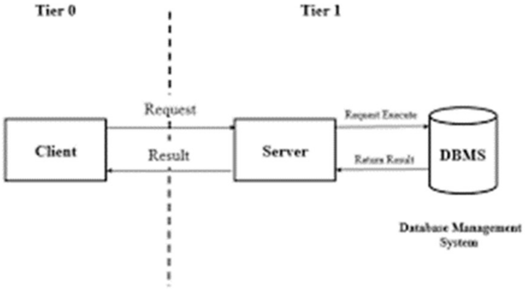


Fig. 1: Role of DBMS in a Two-Tier Application

Some of the data management tasks that can be performed with the help of a DBMS are as follows:

i. Defining the database: This involves specifying the structure of data to be stored in a database. This can be done by specifying the data type, width of the data field, constrains to be imposed on data in a database.

ii. Storing and manipulating data in a database: Data can be stored in a database by inserting new data or updating the existing data in a database. Removal of undesired data can also be done in a database. Retrieving data already stored in a database can be done by querying the database.

iii. Sharing the database by multiple users: Sharing allows multiple users to access the data available in a database without any conflict. Maintaining the consistency of data in a database while sharing, is taken care by the DBMS by providing locks on data that are being accessed and updated by multiple users in a multi-user environment.

II. UNSTRUCTURED AND STRUCTURED DATA

Prior to the introduction of Database System, data were stored in files maintained by the Operating System. One or more application programs process the data stored in files to produce required information. For instance, in a banking application, data about the customers and their savings accounts are stored in files so that the following operations can be done using the data at a later stage:

1. Debit or credit some amount with an account
2. Finding the balance of an account
3. Generate monthly statements of accounts

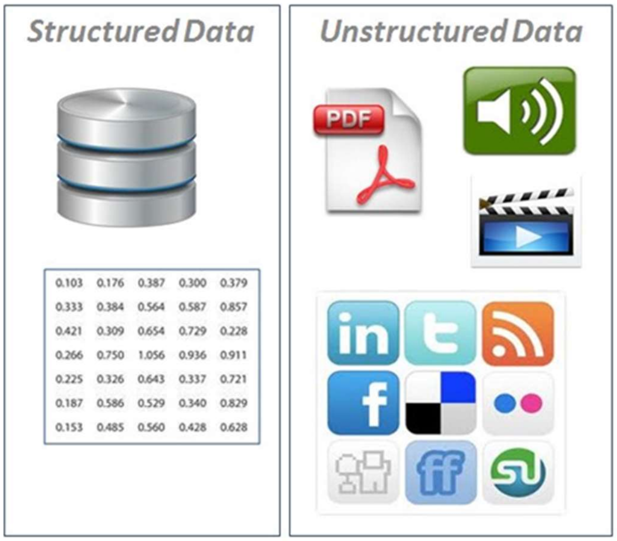
The following are some business applications that make use of databases managed by a DBMS:

1. Automated Teller Machine (ATM) or Online Banking for money transaction
2. Online Reservation system for scheduling and reserving tickets for a travel
3. University or College ERP for registering and pursuing a course of choice
4. Online Book store for searching and buying books online

Files are of various types, which include Text (.txt) files, Rich Text Format (.rtf) files, Portable Document Format (.pdf) files, Hyper Text Document (.html) files, and Document (.doc) files. When data are stored in files, they are unstructured in nature. This is due to the fact that files don't have a structure to follow for storing and organizing data in memory. Application software responsible for creating such files makes use of API calls to OS, which in-turn will perform the necessary disk operations for storage and retrieval of data from memory. By default, data are stored and retrieved sequentially in memory for text files.

In general, text files act as a container for storing plain text in a sequence of lines. Each line of characters is formed using a set of words. Each word may be formed using alphabets, numbers and special characters. Data can be stored and retrieved sequentially one character at a time in text files. Fetching the data randomly is not possible with text files. Hence, storage and retrieval of data in files leads to unstructured data that can't be processed efficiently. Moreover, in a file-processing system, data are stored in one or more files. It is the duty of the application program that accesses those files to define the structure of data to be stored in the file and to manage the file. Following are some of the disadvantages of file systems that are overcome by the database system:

- Data stored in file systems might be redundant and inconsistent
- Related data are isolated from each other by storing them in different files at one or more locations
- Maintaining the integrity of data has to be done at program level
- Automicity of data are not easy to be managed in file system
- Consistency of data may not be possible in a multi-user environment



0.103	0.176	0.387	0.300	0.379
0.333	0.384	0.564	0.587	0.857
0.421	0.309	0.654	0.729	0.228
0.266	0.750	1.056	0.936	0.911
0.225	0.326	0.643	0.337	0.721
0.187	0.586	0.529	0.340	0.829
0.153	0.485	0.560	0.428	0.628

Fig. 2: Structured Vs Unstructured Data

Data Redundancy and Inconsistency: Storing data of an enterprise in flat files may result in data redundancy, i.e., the same information may be stored in multiple files. For instance, when two different files are used for storing the details of Savings Account and Current Account of a customer, the address and phone number of that customer might be stored in both files.

Data Isolation: Data belong to an enterprise are stored in multiple files and each file may store the data in different formats. This makes it difficult to retrieve the appropriate data in a convenient and efficient manner.

Integrity Problems: Integrity of data stored in a file is maintained by enforcing certain types of consistency constraints at the code level. For instance, maintaining non-negative balance of a bank account or having unique values in certain fields such as Account Number are enforced by the programmer through code. On the other hand, database systems provide the mechanism for enforcing such constraints at the database level itself. Hence, data integrity is maintained at the database in database systems.

Automicity Problems: Automicity refers to the consistency of data in a database even after a failure in database transaction that may occur due to hardware or software malfunctioning. It is difficult to achieve automicity if the data are stored in files. But, this is possible by ensuring that the changes made to a file must happen in its entirety or should not happen at all in a database through Transaction Control mechanism.

Concurrent-access Anomolies: Concurrent access of data refers to accessing the data stored in a repository by multiple users at a time. When a file is being accessed by multiple users for storage and retrieval, it may lead to the inconsistency of data when more than one user wants to modify the data at a time. Managing the consistency of data in a file-system is difficult as it doesn't provide the lock necessary to impose on data to ensure the integrity of data when accessed by multiple users. On the other hand, database systems provide the necessary locks on records being accessed by multiple users.

Database Management Systems (DBMS) were introduced to overcome the limitations of file systems. Data are organized in a database in the form of tables consist of rows and columns. The following are some of the characteristics of Database Systems:

1. The self-describing nature of the database system allows meta data to be stored in the database
2. Provides insulation between programs and data by providing data abstraction
3. Supports multiple views of data stored in a database at the view level
4. Allows data to be shared among multiple users in a multi-user environment

Here is a list of functions performed by a Database Management System:

1. Provides persistent storage for data being processed by one or more application programs
2. Restricts unauthorized access from accessing the database
3. Represents complex relationships among data stored in a database
4. Controls redundancy of data in a database by enforcing integrity constraints on relations stored in a database
5. Provides efficient query processing for speeding up the retrieval of data from the database
6. Manages backup of data available in a database for recovery of data during system failure
7. Supports multi-user view on data through data abstraction

Structured Query Language (SQL) is the language understood by DBMS for defining and manipulating data in a database. The statements used in SQL are non procedural, i.e., SQL statements require a user to specify what to be done on data without specifying how to get the work done on the data. Users of database make use of some form of interface to interact with DBMS directly or indirectly. The Database Administrator directly works with the DBMS and interacts with it through DDL (Data Definition Commands) commands or Privileged Commands. DDL Statements given by a DBA will be processed by DDL Compiler to structure the database (schema) by storing the metadata in the System Catalog.

III. FINDINGS FROM EXISTING LITERATURE

There are three stages at which literature review has to be done during the life cycle of a research work:

1. In the early days of the research work, a preliminary research has to be done in order to identify the context (specific area) of the research work.
2. As the research work progresses, the researcher has to read through more literature in order to find the research gap in his/her area of research work
3. In the third stage, the researcher has to read through many more literature or dissertation to relate his/her findings with that of other

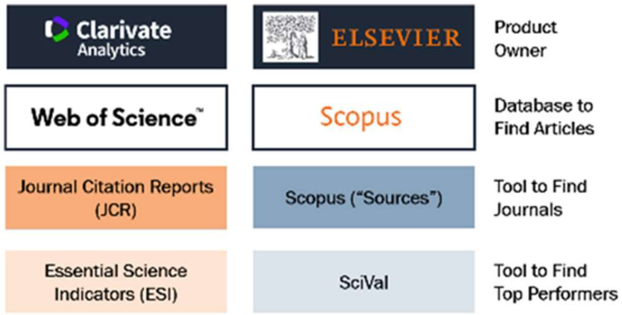

Clarivate Analytics
ELSEVIER
Product Owner
Web of Science™
Scopus
Database to Find Articles
Journal Citation Reports (JCR)
Scopus ("Sources")
Tool to Find Journals
Essential Science Indicators (ESI)
SciVal
Tool to Find Top Performers

Fig. 3: Major Databases and Tools for Reviewing the Literature

Hassan et al (2021) in their survey paper on Servlerless Computing identified five database sources available for exploring the scholarly literature. The database sources used by them for doing literature review in their research work include the following: IEE Explore, Elsevier ScienceDirect, ACM Digital Library, Scopus and SpringerLink [1].

David Livingston et al (2019) in their journal article identified the need for Client/Server model of Encryption in which Client-side encryption takes place at the client side using client-side scripts stored and executed on a browser and server-side storage and retrieval takes place at the back-end on the server using a Database Server like MySQL in an encrypted format [2]. David Livingston et al (2021) in their conference paper identified that the cloud computing follows the architecture of distributed computing in which data are not stored in one particular computer but distributed and stored in multiple computers that are running simultaneously at various data centers. And hence, data stored in a cloud environment are called distributed data, because they are stored and retrieved from distributed databases. They also identified data privacy as one of the major challenges that must be dealt with before moving the sensitive data onto the cloud. They concluded that the data must be encrypted at the client side using any one of the Symmetric Cryptographic algorithms before their migration to the cloud. They introduced two versions of extended play-fair cryptographic algorithms for improving the data privacy of cloud data at the client side [3].

David Livingston et al (2022) in their conference paper identified data in three levels of confidentiality while storing and managing data in a centralized location : Level 1, Level 2 and Level 3. Level 1 refers to data that are critical and having higher level of confidentiality. Level 2 includes data that are important but are not critical in terms of confidentiality. The data identified in Level 3 are not important as well as critical and hence they are less confidential in an organization. They suggested two different symmetric cryptographic algorithms for dealing with Level 1 and Level 2 data stored in a file or database. Block Cipher such as Advanced

Encryption Standard (AES) is suggested for managing the data having Level 1 confidentiality, whereas data in Level 2 can be stored and retrieved using extended Play-fair algorithm introduced by the same author [4].

Jesus Carretero et al (2014) analyzed the tools and techniques used in Cloud Computing for resource sharing [5]. Mazhar Ali et al (2015) addressed the security issues that arises at various levels of Cloud Computing [6]. Rania Fahim El-Gazzar (2016) in their conference paper identified the risks faced by SMEs while adapting Cloud Computing services for an organization [7]. Vasundra Arora (2012) suggested a scheme for searchable encryption of cloud data in order to have the ability to search through the encrypted data in a ranked manner [8]. Sabiyyah Sabir (2018) in his review paper observed that combination of both client-side and server-side encryption algorithm can be used for strengthening the confidentiality of data having highest level of confidentiality (Level 1) on the server [9].

Vishar Kher et al (2005) reviewed the security services required for securing the storage file systems. They identified the need for end-to-end encryption for improving the confidentiality of data from unauthorized users in a multi user environment. They also addressed the need for key sharing and management through which secret key can be shared securely with those who want to decrypt the cipher text that have been encrypted using any one of the symmetric cryptographic algorithms [10]. Abadi D J (2009) in his paper explored the advantages and disadvantages of deploying database systems in the cloud. He concluded that cloud is suitable for deploying databases containing data that are read often for analytical purposes rather than for storing data that are transactional in nature [11].

IV. CONVERTING UNSTRUCTURED DATA INTO STRUCTURED DATA

Semi-structured data refers to data that are stored in files but in some specific format. Examples for files having semi-structure data include eXtensible Markup Language (.xml) and Comma Separated Values (.csv) files. Semi-structured data can be retrieved and processed from the front-end (client application) using some special tools like XML Parser, Document Object Model (DOM) and File System Object (FSO) provided in higher level languages such as Java, Client-side scripting Languages such as VBScript or JavaScript, Server-side Scripting Languages such as Java Server Pages (JSP) or Active Server Pages (ASP).

XML is a meta-markup language used for describing the structure of data in a hierarchical order using tags, where as CSV is a file format used for representing data separated by comma in a file. XML is a markup language used to describe the structure of data stored in a file with the extension .xml. XML is used for data representation in the following context:

(i) Structuring data in a file where a relational database is inappropriate

(ii) Structuring data that can be presented on a web page

Semi-structured data are stored in files using intermediate form that lies between structured and unstructured form of data representation. Semi-structured data can be easily converted into structured data. For instance, data stored in Excel files are in semi-structured format and hence they can be converted into CSV format before they can be stored in the structured format in a database. The following are some of the characteristics of semi-structured data:

- Semi-structure data are stored in files for data representation using special format (structure)
- Semi-structure data can be accessed and processed using tools such as XML Parser, Document Object Model, File System Object from front-end application

- Semi-structured data can be easily converted to structured data since they are represented using some specific format (structure)
- Semi-structured format can't be used for representing large volume of data as they are stored in files not in databases

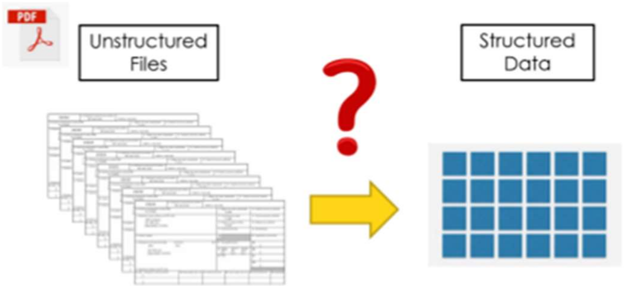


Fig. 4: Framework for Converting Unstructured File into Database Table

Here is a framework that allows a text file to be stored as a table in a Database. By converting an unstructured text file into a table of textual data in a database, we make the data structural and stored in a set of rows that can be accessed randomly. In this framework, the conversion of unstructured data from a file into a structured format (table) in a database takes place as follows:

(i) A Table with four attributes - Id, txtData, lineCount, wordCount needs to be created in the database for storing the content of the text file in a structured format and let the table be named after the name of the text file.

(ii) Store the content of the text file into the table created for it by reading the file line by line and word by word and create tuples by inserting a new record for each and every piece of text read from the text file. Also store the word count and line count of the particular piece of text in order to keep track of the location of the data in the file.

(iii) Apply cryptography using Advanced Encryption Standard (AES) algorithm before storing the data in tuples of the table created for the text file in the database.

(iv) Storing the content of a text file in a database has to be done by storing the content of the text file in multiple rows of a table so that data retrieval can be done using SQL Select Query. When searching for some piece of text, the search string must be converted to cipher text using AES algorithm and then used as a search string in the SQL Query. In this framework, the symmetric algorithm AES will act as a searchable symmetric algorithm that can search a piece of encrypted text stored in a database without decryption process.

V. CONCLUSION

In this research, the authors have identified the need for securing the privacy of structured and unstructured data on the cloud. As the data are to be stored and maintained in the cloud, especially in public cloud, they are distributed in nature. Thus the cloud data must be encrypted before their migration onto the cloud. For securing the privacy of important and critical data (level 1), block cipher such as Advanced Encryption Standard (AES) is recommended. The framework newly introduced in this paper allows unstructured data stored in text files to be converted into structured data that can be stored and retrieved from a database. This framework not only allows the unstructured data to be stored in a database encrypted but also allows the SQL query to search through the data even though they are in an encrypted format.

REFERENCES

[1] H. B. Hassan, S. A. Barakat, and Q. I. Sarhan, “Survey on serverless computing,” Journal of Cloud Computing, vol. 10, no. 1. Springer Science and Business Media Deutschland GmbH, Dec. 01, 2021, doi: 10.1186/s13677-021-00253-7.

[2] David Livingston J, Kirubakaran E, “Client/Server Model of Data Privacy Extended Playfair Cipher for SaaS Applications on the Cloud”, International Journal of Innovative Technology and Exploring Engineering, August 2019, DOI:10.35940/IJITEE.j9274.088101, https://www.ijitee.org/wp-content/uploads/papers/v8i10/J92740881019.pdf

[3] David Livingston J., Kirubakaran E. (2021) Implementation of Extended Play-Fair Algorithm for Client-Side Encryption of Cloud Data. In: Peter J., Fernandes S., Alavi A. (eds) Intelligence in Big Data Technologies—Beyond the Hype. Advances in Intelligent Systems and Computing, vol 1167. Springer, Singapore. https://doi.org/10.1007/978-981-15-5285-4 48

[4] David Livingston, J., Kirubakaran, E., Immanuel Johnraja, J. (2023). Implementing Client-Side Encryption for Enforcing Data Privacy on the Web Using Symmetric Cryptography: A Research Paper. In: Goyal, D., Kumar, A., Piuri, V., Paprzycki, M. (eds) Proceedings of the Third International Conference on Information Management and Machine Intelligence. Algorithms for Intelligent Systems. Springer, Singapore. https://doi.org/10.1007/978-981-19-2065-3-2

[5] Jesus Carretero, Javier Garcia Blas, Introduction to Cloud Computing: Platforms and Solutions, 2014, Journal of Cluster Computing, Published online: 15 February 2014, DOI 10.1007/s10586-014-0352-5

[6] Mazhar Ali, Samee U. Khan, Athanasios V. Vasilakos, “Security in Cloud Computing: Opportunity and Challenges”, Journal of Information Sciences, 2015, Pp. 357-383, http://dx.doi.org./10.1016/j.ins.2015.01.025

[7] Rania Fahim El-Gazzar, “A Literature Review on Cloud Computing Adoption Issues in Enterprises”, 2016, HAL Id: hal-01381189, https://hal.inria.fr/hal-01381189

[8] Vasundra Arora, “Synopsis on Searchable Encryption and Data Retrieval in Cloud Computing”, 2012, Department of Computer Science and Engineering, Manav Rachna International University.

[9] Sabir, S., 2018. Security issues in cloud computing and their solutions: a review. International Journal of Advanced Computer Science and Applications, 9(11).

[10] Kher, V. and Kim, Y., 2005, November. Securing distributed storage: challenges, techniques, and systems. In Proceedings of the 2005 ACM workshop on Storage security and survivability (pp. 9-25).

[11] Abadi, D.J., 2009. Data management in the cloud: Limitations and opportunities. IEEE Data Eng. Bull., 32(1), pp.3-12.

[12] Merkle, Ralph C. "Protocols for public key cryptosystems." 1980 IEEE symposium on security and privacy. IEEE, 1980.

[13] E. P. A. David Livingston J, Kirubakaran E, “Implementing Data Privacy of Cloud Data on a Remote Server using Symmetric Cryptographic Algorithms,” International Journal of Advances in Data and Information Systems, vol. 2, no. 1. Indonesian Scientific Journal, Indonesia, Coimbatore, pp. 62–72, 2021, doi: 10.25008/ijadis.v2i1.1217.

[14] M. A. M. A. Elsayed, “Advancing Security Services for Cloud Applications,” ProQuest Diss. Theses, no. September, p. 225, 2018

[15] A. A. Omotunde, A. A. Izang, O. . Awoniyi, B. K. Omotunde, and Y. A. Mensah, “Cloud Computing Awareness and Adoption Among Small and Medium Scale Businesses,” Int. J. Multidiscip. Sci. Eng., vol. 6, no. 6, pp. 32–38, 2015. https://doi.org/10.2139/ssrn.3305955

[16] P. R. Kumar, P. H. Raj, and P. Jelciana, “Exploring Data Security Issues and Solutions in Cloud Computing,” in Procedia Computer Science, 2018, vol. 125, pp. 691–697, doi: 10.1016/j.procs.2017.12.089.

[17] M. SarfarazIqbal, S. Singh, and A. Jaiswal, “Symmetric Key Cryptography: Technological Developments in the Field,” Int. J. Comput. Appl., vol. 117, no. 15, p. 4, 2015, doi: 10.5120/20631-3248.

[18] S. Bhattacharya, “Cryptology and Information Security-Past, Present, and Future Role in Society,” Int. J. Cryptogr. Inf. Secur., vol. 9, no. 1, pp. 13–36, 2019, doi: 10.5121/ijcis.2019.9202.

[19] S. Tahir, S. Ruj, Y. Rahulamathavan, M. Rajarajan, and C. Glackin, “A New Secure and Lightweight Searchable Encryption Scheme over Encrypted Cloud Data,” IEEE Trans. Emerg. Top. Comput., vol. 7, no. 4, pp. 530–544, 2019, doi: 10.1109/TETC.2017.2737789.

[20] I. O. Adam and A. Musah, “Small and Medium Enterprises (SMEs) in the Cloud in Developing Countries: A Synthesis of the Literature and Future Research Directions,” J. Manag. Sustain., vol. 5, no. 1, pp. 115–139, 2015, doi: 10.5539/jms.v5n1p115.