International Journal of Advances in Data and Information Systems
Vol. 2, No. 1, April 2021, pp. 62~72
ISSN: 2721-3056, DOI: 10.25008/ijadis.v2i1.1217

Implementing Data Privacy of Cloud Data on a Remote Server using Symmetric Cryptographic Algorithms

David Livingston [1], Ezra Kirubakaran[2], Eben Priya David [3]
[1,2] Department of Computer Science & Engineering, Karunya Institute of Technology, Coimbatore, India.
[3] Department of Management Studies, Karunya Institute of Technology, Coimbatore, India.

Article Info

Article history:

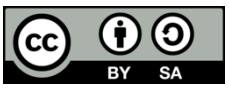
Received Feb 26, 2021
Revised April 3, 2021
Accepted April 20, 2021

Keywords:

Cloud Computing
Micro Small and Medium Enterprises
Information Technology
Cloud Service Providers
Cryptographic Algorithm
Advanced Encryption Standard

ABSTRACT

Cloud Computing is an excellent technology for Micro Small and Medium Enterprises, which operate under budget shortage for setting up their own Information Technology infrastructure that requires capital investment on resources such as computers, storage and networking devices. Now-a-days, major Cloud Providers like Google and Amazon provide cloud services to its customers for managing their email, contact list, calendar, documents, and their own websites. MSME can take advantage of the cloud-based solutions offered by various Cloud Service Providers for equipping their own employees in doing their day-to-day activities more effectively and on the cloud. Though cloud computing promotes less expensive and collaborative work environment among a group of employees, it involves risks in keeping the resources such as computing and data secured. Different mechanisms are available for securing the data on the cloud among which encryption of data using cryptographic algorithm is the widely used one. Among various symmetric cryptographic algorithms, Advanced Encryption Standard is the more secured symmetric algorithm for implementing data privacy on the cloud. In this paper, the authors have discussed some of the issues involved in adopting the cloud in an organization and proposed solutions that will benefit an organization while uploading and managing data in files and databases on the cloud.



Corresponding Author:

David Livingston, Research Scholar,
Department of Computer Science and Engineering,
Karunya Institute of Technology and Sciences,
Karunya Nagar, Coimbatore – 641 114, Tamilnadu, S. India.
Email: davidjlivingston@gmail.com

1. INTRODUCTION

In cloud computing, virtualization enables centralized management of pooled resources such as storage and computation. It allows virtual resources created and provisioned from a pool of physical resources. From a public cloud perspective, virtualization appears as a shared resource (such as OS, storage, database, application) at various layers of the virtualized service. For instance, storage virtualization is a kind of virtualization that enables multiple pooled storage devices appear together as a single large storage entity. As per the online reference Wikipedia, Infrastructure-as-a-Servier (IaaS) is a delivery of computer infrastructure (typically in a virtualized environment) as a service. Platform-as-a-Service (PaaS) provides a platform on top of IaaS for building and running custom web-based application.

Software-as-a-Service (SaaS) is a software distribution model in which applications are hosted and made available to customers over the Internet. In the public cloud model, the critical information of an organization is generally stored and maintained by Cloud Service Providers (CSP) at various geographical locations in the cloud platform. The organization does not have any control over the data on the cloud. Though Cloud Service Providers such as Amazon and Google provide their own mechanism for securing resources stored and maintained on the cloud, it is the sole responsibility of the Cloud Users to secure their own resources especially sensitive data on the cloud storage.

According to the report prepared and submitted by Mr. Holger Schulze, CEO and Founder, Cyber Security Insider, the biggest challenge any organization can face while adopting the cloud is data security, data loss and data leakage. Cyber Security professionals make use of mechanisms like access control (52%), encryption or tokenization (48%), security services offered by the cloud providers (45%), cloud security monitoring tools (36%), and connection using protected networks (36%). Based on this report, we can come to know that the encryption or tokenization is the second widely used method of data privacy [1]. Thus, the data privacy on the cloud can be achieved by using one of the symmetric cryptographic algorithms like AES (Advanced Encryption Standard).

In the previous paper written by the same author, the authors have come up with a client/server model for preserving the privacy of data on the cloud [2]. In that paper, they have argued that the sensitive data must be encrypted first using one of the symmetric cryptographic algorithms before moving them on to the cloud. At the client side itself, the data must be secured by encrypting them using a symmetric encryption algorithm like AES (Advanced Encryption Standard). They have also suggested a naming convention for naming the columns (attributes) of tables (relation) used for storing data in a database, so that intruders can't easily identify the usage of each and every field (column) of table data.

In another paper presented at an international conference, they have introduced an extended algorithm by modifying play-fair algorithm that can be used for performing searchable encryption on cipher text stored in a cloud database [3]. The objective of the research work carried out and presented in this paper is to implement data privacy on the server while performing the following operations: (i) uploading and managing text files on the public cloud, (ii) importing data from excel files to a database on the cloud and (iii) encrypting data in files/database after storing them on the cloud.

2. CLOUD SOLUTIONS FOR COLLABORATIVE WORK

Cloud makes it easy to communicate and collaborate with peers. Some of the cloud solutions (also called as Software as a Service) available for collaboration among peers for sharing resources and ideas. With Google Docs, multiple people can work at the same time on the same document, and each person can see the changes made by others as soon as they save their work [4]. Google Calendar lets the user organize his/her schedule and share events with co-workers. With this Google App, it is easy to keep track of our daily schedule. Google Contacts, the address book manager of Gmail lets the user keep track of all the contacts, which include the contact details of colleagues and customers.

With the help of search engines such as Google Scholar and Microsoft Academia, research scholars can search for literature on the web. Search engines help researchers search for existing literature on the web and to do reference management. Cloud Solutions like Paperria, Overleaf and LaTex Base are available for drafting. These online tools make collaborative writing simple and effective. There are some scientific computing and analysis tools available on the cloud namely

GNU PSPP, SciLab, and Weka. GNU PSPP is a popular tool for statistical analysis. SciLab tool is associated with scientific computing problems. Weka is for data mining tasks.

Cloud storage services such as Google Drive and One Drive are available for storing and sharing files among a group of users on the cloud. Simple Storage Service (S3) is a web service provided by Amazon Web Services for storing and retrieving files from the cloud. Elastic Block Storage (EBS) is another storage service for storage and retrieval of bulk data on the cloud. AWS also provides services like RDS (Relational Data Storage) for storing and retrieving data from a relational database [5].

Software as a Service (SaaS) applications are running on top of Platform as a Service (PaaS), which include Operating System and other development tools required for developing the SaaS applications. SaaS Applications that are developed using Open Source Software are also known as Open SaaS Applications. Open Source Software (OSS) is a kind of software which operates under open and free software licensing. Contrary to the proprietary software, which needs licensing from its vendor for its installation and running on any computer, OSS does not require individual licensing for its use. Hence, it eliminates the cost involved in purchasing the software licensing for its installation and usage. This kind of software when used on the cloud for the development of SaaS Applications, it further reduces the upfront cost of developing and deploying SaaS applications. LAMP (Linux Apache MySQL and PHP/Python) stack is an example of Open Source Software. With the help of PHP and MySQL, developers can develop Web Applications that can run on the Internet or cloud for creating dynamic web pages. In this research, Open Source Software - MySQL and PHP are used for implementing data privacy of cloud data on the server.

3. RESEARCH METHODOLOGY

Researchers should be well acquainted with research methodology – the systematic procedure to be adopted in a research work. They have to thoroughly review the existing literatures which are relevant to their research problem so that they can state the main research problem clearly and unambiguously. The problem statement must be formulated in two or three sentences. Every study resolves around a problem and aims at solving it. The researcher has to identify the purpose and scope of the research first. The purpose defines the goal to be achieved, whereas, the scope sets the boundaries for the study and determines the length or volume of the research report.

In any research work, the researcher is expected to provide evidence of reading a certain amount of relevant literature in order to have some awareness of the current state of knowledge of the subject. Conducting a literature review is a skill that can help a researcher to showcase his talents of understanding, interpreting, analyzing, and synthesizing a real-world problem in the area of his research. It enables a researcher to clarify his own thoughts about a particular study and to critically review the relevant literature in order to identify a gap within the literature. The outcome of a literature review is to ‘address a gap’, and not to ‘fill a gap’ in the area of research.

The authors have done a preliminary literature review on the selected area of research – Cloud Computing. As a result of the preliminary review of literature, they have chosen Public Cloud Model as the specific area of research. Within the Public Cloud, SaaS (Software as a Service) deployment model of cloud delivery has been chosen for conducting the research. After choosing a particular area of research - *SaaS in Public Model of Cloud Computing*, the authors have narrowed down their research further to deal with a specific topic - *Information Security on the Cloud.* Hence in this research, the authors have dealt with *Data Privacy on Public Cloud* - one

of the major issues that must be taken care while adopting the public cloud for running a SaaS application in an organization.

Figure 1: Selection of Topic for the Research

Based on the literature review of existing literature, cryptography has been chosen as the standard mechanism of implementing data privacy on the cloud [21]. From the two major classification of cryptography, Symmetric Cryptography has been chosen for encryption of cloud data, due to the fact that it is simple and faster than the Asymmetric Cryptographic approaches [22]. Within the classification of symmetric cryptographic algorithms, Advanced Encryption Standard (AES) is the best one for encrypting sensitive data. For encrypting non-sensitive textual data, the authors have proposed a modified version of play-fair algorithm, using which some of limitations of AES algorithm can be overcome.

4. REVIEW OF LITERATURE

Alves et. al. [6] have proposed a framework for database encryption that preserves data secrecy on an un-trusted environment while performing searching on cloud database. They have employed order revealing encryption to perform selection operation on databases, and homomorphism to enable computation over cipher texts [6]. L. Arockiam et al (2013) have analyzed the issues involved in managing data privacy on the cloud. They have proposed a technique that improves the mechanism used in classical encryption algorithms such as substitution cipher and transposition cipher. Both substitution and transposition techniques use alphabet for generating cipher text. Whereas, in the new approach proposed by them, the plain text is converted into ASCII code and then the ASCII code is converted into cipher text [7]. Keiko Hashizume et al (2013) have identified that data leakage happens when the data gets into the wrong hands while it is being transferred, stored, audited or processed. They recommended encryption as the mechanism for securing sensitive data [8].

Ming Li et. al. [9] have mentioned about end-to-end encryption as an approach for having control over the data on the cloud. They identified the need for Searchable Encryption (SE) technique, which ensures privacy-assured search over encrypted data. They advocated that two key properties – function usability and efficiency – must be achieved in order to build privacy-assured searchable cloud data services. They introduced a methodology and described the building blocks required to design usable and efficient privacy-assured search schemes on encrypted data. They concluded that the methodology they propose performs functionality rich, usable and efficient search on encrypted data without sacrificing privacy too much [9].

Salam et. al. [10] have tried to implement a privacy preserving data storage and retrieval system. In this approach, they have chosen one of the symmetric key primitives due to its efficiency in mobile environments. This new approach enables its users to search over the encrypted data without revealing any information about the data or the enquiry [10]. Dan Boneh et. al. [11] in their journal article showed that using a polynomial encoding of the database enables efficient implementations of conjunction queries using somewhat homomorphic encryption. They

presented two implementations of their protocol - using Paillier‟s additively homomorphic system as well as Brakerski‟s somewhat homomorphic cryptosystem [11].

Kariti et, al. [12] have proposed a privacy preserving architecture, which is based on encryption to protect the data on the cloud from unauthorized access. In this architecture, they use AES as the symmetric-key algorithm for implementing data privacy on the cloud. They also identified the need for changing the symmetric key frequently sothat the data privacy of cloud data can be extended further [12]. Kefa Rabah [13] has analyzed Data Encryption Standard (DES) algorithm and its sustabinability in securing data. He has compared the performance of DES with that of the other symmetric key cryptsystems. He suggested that cryptographic algorithms must be carefully selected and implemented in order to have a strong cryptosystem that can maintain the secrecy of data in transit or at rest [13].

Solomon et. al. [14] in their journal article presented a framework called ZeroVis framework that provides confidentiality for data stored in a cloud environment. In this framework, they provided fine-grained access control with the ability to search over encrypted data so that existing applications can get migrated to cloud environments with very minimal software changes [14]. Rashmi et. al. [15] have found out that the data at rest in storage services such as Amazon Simple Storage Service (S3) is not encrypted by default. Therefore, users need to encrypt their data before it is uploaded to the cloud, so that it is not accessed or tampered with by any unauthorized party. They also suggested that SaaS vendor must adopt additional security checks to ensure data security and prevent breaches due to security vulnerabilities in the application or through malicious employees [15].

Sood [16] have proposed a combined approach for data protection that utilizes various measures such as SSL (Secure Socket Layer) 128-bit or 256-bit encryption (if needed) and MAC (Message Authentication Code) for implementing data integrity and searchable encryption. Other methods identified by them for data security include strict authentication, digital signatures, sensitivity rating and building of index [16]. Arora [17] in his short synopsis submitted for the research he undertook on “Searchable Encryption on Data Retrieval in Cloud Computing” proposed a mechanism in which both encrypted data and the encrypted index file are uploaded on to data centers in the cloud. The cloud server performs the search using the cipher text given by the user on the encrypted index and returns a list of relevant files containing the required data. As a result of this search operaion on the list of files having encrypted data, the required file will be sent to the client [17].

Arshad et. al. [18] have proposed an algorithm that improves the weakness of the original affine cipher. They have discussed some of the cryptography algorithms that can be used to provide security and confidentiality of data in the database. In the new encoding scheme named Enhanced Affine Block Cipher, sensitive data is encrypted outside the database (application level) and then stored in the database [18]. Jaffry [19] in his article written and published on AWS Database Blog explained two different database services provided by Amazon Web Services (AWS) namely Amazon RDS and DynamoDB. He identified that data encryption and tokenization are the two different methods of protecting cloud data against the risk such as data exfiltration through unauthorized access mechanisms [19].

Boldyreva et. al. [20] in their article published in Network Security magazine discussed the capabilities of various schemes available for protecting data on the cloud via company-owned encryption keys while still preserving the functionality of the underlying cloud applications. They

concluded that in a cloud computing environment, it's critical to understand how to weigh security against access trade-offs while choosing a suitable encryption scheme for enforcing data privacy [20].

5. FILE UPLOADING AND MANAGEMENT ON THE SERVER

In this research work, the primary concern is to upload the files and data securely onto the server machine which is maintained by a service provider on the internet or public cloud. By default, a User Interface (UI) is provided by the service provider of the public cloud or domain space, for uploading and managing the resources on the server. The UI provided for this purpose is known as Dashboard or Control Panel respectively. In order to access the storage space available on the server through such UI, the user has to login first using valid credentials to prove him as a valid user. Then only he/she will be allowed to manage the resources on the server. Allowing everyone to access the Dashboard or Control Panel is not a group practice in a collaborative environement. Because, it gives the user the access to all the resources uploaded and managed on the server that includes databases. Hence, we need a tool such as *filegator* – an Open-Source Software for file uploading and downloading from a particular folder on the server.

With the help of this OSS, users can login to upload and manage files in a particular folder on the server. The OSS – *filegator* can be used as such or can be customized using the source code that comes along with the product. The following are few suggestions that are to be considered before its use in our web application or SaaS application on the cloud:

1) The *filegator* uses a subfolder named *repository* under its main folder for file storage and management. This restriction needs to be removed, so that it may allow any folder having any name to be used as the folder for storing and retrieving the shared files.
2) By default, the user credentials – the user name and the password of the users – are stored in a file (users.json) under the folder named *private*. This can be modified in such a way that the user credentials are stored in a table of a database and not in a file. Though the user credentials are stored in a file, passwords of the user are stored in encrypted form for privacy.
3) Opening a file, especially a text file has some problem in *filegator*. Opening a simple text file doesn't take long time to load and display its content in a default text editor of *filegator*. Even in displaying the content of a simple text file, there exists a problem – the content is displayed along with some html tags and some error messages at the top and bottom of the file. When opening a large text file of size more than 1 MB, *filegator* takes long time to load and view its content. This problem can be rectified by providing an option for opening text files using a link on the web page itself with the help of *href* property of <a> tag in html.

There is another Open-Source Software named *ProjectSend* that works similar to that of *filegator* ProjectSend OSS performs most of the operations on files that include uploading and downloading files from the server, deleing files on the server. It also allows its user to compress a set of files before downloading. Unlike *filegator*, *ProjectSend* makes use of a table in a database for storing and managing the credentials of its user. The major limitation of using this project is that it works only on one folder. For storing and managing files on the server, it makes use of a subfolder named *Upload*. Hence, this OSS project is best suited for collaborative file sharing among a group of users. It gives us more control on the files that are stored and managed in a shared folder on the server.

Though *ProjectSend* operates on only one folder, it keeps track of all the activities on files by maintaining a log file. It allows uploading of single files as well as multiple files on to the server. But it doesn't have the provision for extracting the content of a compressed uploaded file on the server, which is required for uploading nested files and folders. The following table list out major differences between the file sharing OSS – *filegator* and *ProjectSend*:

Table 1: Comparison of OSS – filegator and ProjectSend

Feature	OSS Filegator	OSS ProjectSend
Folder used for Storage	*Repository* subfolder	*Upload* subfolder
Resources Managed	Files and Folders	Only files in *Upload* folder
Placeholder for User Credentials	File named *users.json* in the *private* subfolder	Table named *tbl_users* created in MySQL Database
Short Comings	Opens text files using an editor. But, has difficulty in opening text files of size more than 1 MB	Text editor is not provided for opening a text file. Extraction of compressed files can't be done on the server.
Additional Features	Extraction of files stored in a compressed uploaded file is possible.	Operations performed on files by various users are recorded in a log file.
Usage	Software Developers can collaborate with each other in managing files during the development of a project.	End Users can collaborate with each other for storing and sharing files on the server with the help of activity log maintained on the server.

For an effective file sharing operation on the server, it is advisable to make use of both OSS namely *filegator* and *ProjectSend*. *Filegator* is a file manager that overcomes some of the short comings of the Control Panel or Dashboard provided by the Service Providers for managing files. Whereas, *ProjectSend* is a file manager that complements the shortcomings identified in *filegator* OSS. With the help of *filegator*, developers can manage their files and folders at the development time of a project, whereas end users can make use of *ProjectSend* for sharing their files on the server.

The platform used for implementing the concepts explained in this research work is https://in.000webhost.com/. This is a public domain provided by a service provider for storing and managing the web resources on the internet. Users are provided with a control panel that allows its users to manage the files and databases that are created or uploaded on to the server. While working with the control panel provided by the service provider of *000webhost*, the author has found few difficulties that are mentioned here:

1. Due to maintenance work, *cpanel* was not available once for login and to access the web resources stored and maintained on the server.
2. Opening the file manager provided on the control panel for uploading and managing files takes many steps, which can be avoided by making use of the OSS – *filegator*.

While extracting the files available in a compressed file on the server, error occurred. *cpanel* provided by *000webhost*, doesn't support extracting a compressed file of large size, without which installation of an OSS is not possible. This shortcoming is overcome with the help of file manager – *filegator* hosted on the server.

6. DATA IMPORT AND DATA PRIVACY ON THE CLOUD

Importing data and maintaining the privacy of data stored in a remote database is an important activity while adopting the cloud in an organization. Importing data stored in a database, especially MySQL database can be done easily. With the help of Export option provided in the database, data can be easily exported as well as imported from one database to another. In this scenario, the source database is called local database and the destination database is known as remote database, which is created and maintained on the cloud or a public domain. When Excel sheets are used for storing the data manually in an organization, exporting them to a remote server that stores its data in a database involves few steps:

1) Convert the Excel file to CSV (Comma Separated Values) file.

2) Upload the converted CSV file on to the Server using control panel or file manager OSS.
3) Create a Table in MySQL for storing the content of the Uploaded CSV file
4) Import the data from Uploaded CSV file to the table in the Database

The first step involved in importing data from excel file is to convert the excel file from .xls format to .csv file format, so that it will be in a readable form on the server. Then the converted csv file must be uploaded to the server for reading and importing data to a database. For uploading and storing the csv file, there is a subfolder named *repository*, where all the uploaded files are stored and managed on the server. It is better to upload the csv file one by one, though bulk upload (uploading more than one file at a time) is possible using the OSS file manager – *filegator*. Once the file has been uploaded and kept on the server, the next step in the import process is to create a table structure in the MySQL database for data storage and retrieval. Table is a placeholder for data and the data are arranged in row and column manner. Tables are also called as relation in a Relational Database Management System such as MySQL. Each row of data is called a *record* or *tuple*, and each *column* refers to an *attribute* that can hold some data of a record.

For each and every comma separated value in the .csv file, separate column (attribute or field) has to be created in the table. At the time of table creation, each column is given a name using the alphabet 'c' and a number (ordinal number that starts with 1). Similarly, each and every table is given a name using the string "tbl" and a number. For instance, *tbl10* is the table name given to a table created after 9 tables which are already created and stored in the database. This is the naming convention used for naming the column and the table while creating a table in the database. Moreover, the data type of each and every column in the table created newly is assumed to be *variable length string (VarChar in Oracle or MySQL)*, so that the columns can accommodate any type of data in them. After creating the table for storing data in MySQL database, the .csv file will be moved from its original position (repository folder) to a sub folder named *Encrypted*, where it will be kept till all its contents are imported to the table in the database. The final step in importing data on the server is to create records in the table of MySQL database for each and every line of comma separated values in the .csv file. After importing the data from .csv file to the table in a database, the .csv file has to be deleted, so that the data can't be stolen and misused by an intruder.

Encryption of data in the database can be done on the server after uploading and while importing the data to the database. Depending on the level of sensitivity of data, encryption must take place at the early stage of importing process. The encryption algorithm chosen for encrypting the sensitive data of a database is Advanced Encryption Standard (AES). For encrypting non-sensitive or searchable data, an extended play-fair algorithm is proposed.

7. SERVER-SIDE ENCRYPTION OF TEXT FILES

For securing the content of the text file on the server, encryption algorithm – Advanced Encryption Standard (AES) is recommended. AES is a kind of symmetric encryption algorithm, which makes use of the same key for encryption as well as for decryption process. While encrypting the contents of a text file, encryption is applied line by line. The user has to provide the key at the time of encrypting the text file. To keep the encrypted file separately, a sub folder named *Encrypted* has been created and maintained on the server. While decrypting the content of an encrypted file, a special file named Encrypted.txt is used in the *repository* (folder used for storing all the uploaded files) folder itself. For viewing the encrypted form (also called as cipher text) of a text file, html link using <a> tag can be provided in such a way that by clicking which the content of the file (cipher text) will be displayed in a separate tab of the browser.

Each and every file to be encrypted and stored on the cloud will use a different key for its encryption process. There is a separate table provided in MySQL database for managing the keys. The keys are encrypted and stored in a table named *tbl2* in the database on the server itself. A special key is used for encrypting the keys that are associated with each and every resource (be it a file or a table in the database) on the server. The special key is used at the time of encryption as

well as decryption. Hence, for encryption, we need two keys: one is the special key (key to encrypt the actual key) and the other one is the key used for encrypting the content itself. But for decrypting the cipher text stored in a file or database, we need only the special key. Because, using the special key, it is possible to obtain the actual key used for decrypting the cipher text. Therefore, it is the duty of the end user to keep the special key secret so that the privacy of data can be maintained successfully on the server.

Encrypting the data needs to be done for text files as soon as they are uploaded. After encryption, they are moved and kept in a subfolder named *Encrypted*. After the encryption process, the original file having the plain text must be deleted from the server. Similarly, data to be stored and kept in a database must be encrypted as soon as they are imported from an external source (.csv file). Importing data from an external source is a onetime process. Once the database is setup and running, the data to be kept in a table must be encrypted at the time of insertion. For encryption and decryption of files on the server, either we can go for the standard algorithm like AES or an extended classical algorithm like extended play-fair algorithm. AES is the best option for securing the data which are more sensitive in nature. But, searching a sub string is not possible when the data are encrypted using AES. This limitation of AES is overcome by using an extended play-fair algorithm, which is meant for encryption of less sensitive data in a file or database.

8. CONCLUSION

In this paper, the authors have identified the need for an Open-Source Software such as *filegator* that can be used for easily uploading and managing the files on the cloud. They have also identified another OSS file manager named *ProjectSend* using which file sharing can be done collaboratively among a group of users. The major differences between the OSS - *filegator* and ProjectSend have been identified and suggestions for improvement are given. In this research, the authors have also found the need for a tool for migrating data from the client onto the server, since the data might have been stored in various file formats that include text, comma separated values (csv) or in Excel (xls) files. Moreover, suggestions were made for using algorithms like Advanced Encryption Standard (AES) for storing sensitive data on the cloud. For searchable encryption, they have proposed an Extended Play-fair symmetric-key algorithm using which data privacy on non-sensitive data can be implemented on the server.

REFERENCES

[1] https://www.synopsys.com/software-integrity/resources/analyst-reports/security-in-the-cloud.html

[2] David Livingston J, Kirubakaran E, "Client/Server Model of Data Privacy using 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, "Implementation of Extended Play-Fair Algorithm for Client-Side Encryption of Cloud Data", Part of the Advances in Intelligent Systems and Computing book series (AISC, volume 1167), Springer, Singapore

[4] https://gsuite.google.com/learning-center/products/docs/get-started/#!/

[5] Overview of Amazon Web Servies, AWS Whitepaper Abstract, Published in April 2017

[6] Pedro G.M.R. Alves, Diego F. Aranha, "A Framework for Searching Encrypted Databases", 2018, Journal of Internet Services and Applications, https://link.springer.com/article/10.1186/s13174-017-0073-0

[7] L. Arockiam, S. Monikandan, "Data Security and Privacy in Cloud Storage using Hybrid Symmetric Encryption Algorithm", 2013, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 8, August 2013, ISSN (Online) : 2278-1021

[8] Keiko Hashizume, David G Rosado, Eduardo Fernandez Medina, Eduardo B Fermandez, "An Analysis of Security Issues for the Cloud", Journal of Internet Services and Applications, 2013, http://jisajournal.com/content/4/1/5

[9] Ming Li, Shucheng Yu, Kui Ren, Wenjing Lou, Y. Thomas Hou, "Towards PrivacyAssured and Searchable Cloud Data Storage Services", IEEE Network, 2013

[10] Md Iftekhar Salam, Wei-Chen Yau, Ji-Jian Chin, Swee-Huay Heng, Huo-Chong Ling, Raphael C-W Phan, Geong Sen Poh, Syh-Yuan Tan, Wun-She Yap, "Implementation of Searchable Symmetric Encryption for Privacy Preserving Keyword Search on Cloud Storage", 2015, Journal of Human-Centric Computing and Information Sciences, DOI: 10.1186/s13673-015-0039-9

[11] Dan Boneh, Craig Gentry, Shai Halevi, Frank Wang, and David J. Wu, "Private Database Queries Using Somewhat Homomorphic Encryption", 2013, ACNS 2013, LNCS 7954, pp. 102–118, 2013

[12] Zaid Kariti, Mohamed El Marraki, "Applying Encryption Algorithm to Enhance Data Security in Cloud Storage", Article published in Engineering Letters · November 2015

[13] Kefa Rabah, "Theory and Implementation of Data Encryption Standard: A Review", Information Technology Journal, 2005, ISSN 1812-5638

[14] Michael G. Solomon, Vaidy Sundera, Li Xiong, "Towards Secure Cloud Database with Fine-Grained Access Control", 2014, International Federation for Information Processing 2014

[15] Rashmi, G. Sahoo, S. Mehfuz, "Securing Software as a Service Model of Cloud Computing: Issues and Solutions", 2013, International Journal on Cloud Computing: Services and Architectures (IJCCSA), Vol. 3, No. 4, August 2013

[16] Sandeep K Sood, "A Combined Approach to Ensure Data Security in Cloud Computing", 2012, Journal of Network and Computer Applications, Available online at http://dx.doi.org/10.1016/j.jnca.2012.07.007

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

[18] Noor Habibah Arshad, Saharbudin Naim Tahir Shah, Azlinah Mohamed, Abdul Manaf Mamat, "The Design and Implementation of Database Encryption", 2007, International Journal of Applied Mathematics and Informatics, Issue 3, Volume 1, 2007

[19] Syed Jaffry, "Best Practices for Securing Sensitive Data in AWS Data Store", 2018, Published on line on December 24, 2018 at AWS Database Blog

[20] Alexandra Boldyreva, Paul Grubbs, "Making Encryption Work in the Cloud", 2014, Network Security, October 2014

[21] Amar Ghorbel, Mahmoud Ghorbel, Mohamed Jmaiel, "Privacy in Cloud Computing Environments: a Survey and Research Challenges", Journal of Supercomputing, 2017, Published online: 23 January 2017, DOI 10.1007/s11227-016-1953-y

[22] Madhumita Panda, "Performance Evaluation of Symmetric Encryption Algorithms for Information Security", 2017, International Journal of Advanced Research Trends in Engineering and Technology, Volume 4, Issue 11, November 217