

Cloud Based File Sharing Service for Enhanced Anonymity

Jitesh Mishra
B.Tech (CsBs)
NMIMS STME
Kharghar, India
jitesh.mishra0210@gmail.com

Sujeet Patil
B.Tech (CsBs)
NMIMS STME
Kharghar, India
sujeetpatil111@gmail.com

Shreya Shetty
B.Tech (CsBs)
NMIMS STME
Kharghar, India
shettyshreya.sj@gmail.com

Yashasvi Thakur
B.Tech (CsBs)
NMIMS STME
Kharghar, India
yashasvi.thakur5001@gmail.com

Abstract - In today's world a lot of focus is given to security and very little to anonymity. There are a variety of contexts in which users have legitimate reasons to want to exchange information, while maintaining anonymity. In many file-sharing services, the users can be tracked. The research methodology for the current paper is a literature survey, and we have developed a file sharing service based on the Python programming language and cloud. A Cloud-Based Architecture is used to assure users' anonymity. Nevertheless, a dual layer of encryption is used to encrypt the object. The AES-256 encryption algorithm is the first layer, in addition to the cloud-based encryption, as the second layer ensures that the file uploaded is kept secure. This paper proposes development of an anonymous file sharing model for hurried sharing of files. The key feature of this service is that there is no user authentication required. However, it requires file-based authentication, providing the user with a secure key to access the file along with password protection.

Keywords - Flask, Cloud Computing, MongoDB, User anonymity, File sharing service.

I. Introduction

Cloud computing is now a major topic due to its numerous benefits, it allows you to connect to the architecture through the internet and use computational resources without having to install and maintain them locally.

The transition from locally installed applications to cloud computing is only just getting started. Organizations that used to utilize the cloud were formerly frowned upon, but

now it has revolutionised a majority of enterprises. According to a recent survey, 92 percent of enterprises are now "partially" cloud-based.

Virtualization is the process that makes cloud computing feasible. Virtualization enables the development of a replicated, online "virtualized" computer that functions similarly to a real computer. Cloud is quickly establishing itself as the sole hosting platform for all sorts of programs and technical developments. Cloud has revolutionized and become an integral part of our lives, with the majority of our day-to-day activities taking place on the cloud.

One of the cloud computing services is Software as a Service(SaaS). SaaS is a software delivery paradigm in which software is acquired as a service through the Internet. There are numerous benefits to using Software as a Service. Operational and innovational advantages can be gained by using SaaS.

Initially, SaaS products were designed to solve a primary business need. Since then, SaaS has seen a number of major changes and is now used by several organisations.

Nevertheless, cloud computing is still a new paradigm that enables a network to quickly obtain access to computer resources with minimum effort.

Recent advancements in cloud computing have resulted in a slew of unanticipated anonymity concerns in many components of cloud systems.

Cloud computing is widely utilized for data storage by users and organizations. The data stored on cloud servers that may be more widely available on a variety of domains and distances. This raises concerns about the security of the data. Additionally, users may want to remain anonymous while performing any transaction in the cloud. In this paper, we introduce an anonymous access provision system utilizing symmetric encryption algorithms. The proposed system avails to ameliorate data security with users and to maintain their integrity and confidentiality without compromising computer intricacy.

Oversight control of access to and security of information is among the key barriers to the inevitably ineluctable adoption of distributed computer services. These services are shared by sundry users and are conventionally available through an informal organization, such as the Internet. The information provided in such a case requires privacy and assurance of trust and access control.

Our cloud framework is awe-inspiring for users who wish to apportion files, yet wish to keep their personality nebulous and uninformed. They may additionally require that information be obtained from a user in particular and that sharing be kept confidential. We consider the anonymity and authenticity of the file sharing process, the anonymity of files by the user, and the anonymity of the information itself. Apart from anonymity, authentication quandary is consequential in the case of file sharing.

Regarding above challenges problem statement of this paper is:

“Develop a cloud based SaaS system to promote user anonymity without breach of information.”

This paper talks about development of service based on Flask which is a micro web framework written in Python which acts as a primary backend and is responsible for functioning of the service i.e . serving pages as a response. Flask is also a very popular web server and this is the reason why we are choosing to use it for the frontend. The web interface is used for interacting with the backend in real time. Therefore, it is necessary to have a backend code in place which should be running at a high level.

MongoDB is a source-available cross-platform document-oriented database program responsible for maintaining records of both files and users using the service . MongoDB makes it possible to access and manipulate the information of any file with the aid of a simple scripting language. The database is available on Windows, Mac OS X and Linux operating systems . Mongo is used in many types of companies to store information in the form of documents in which documents contain collections of data.

The Amazon Web Services (AWS) is a collection of remote computing services that together make up a cloud-computing platform, offered over the Internet by Amazon.com. The most central and well-known of these services is Amazon EC2. AWS is one of the leading cloud providers in the world, with common usage being IT infrastructure for website hosting, storage platforms, and content delivery networks.

AWS and Azure are our cloud providers responsible for hosting the service on an EC2 instance of AWS and the storage of files in blob storage of Azure.

II. LITERATURE SURVEY

This Literature survey indicates that very few studies have been published with the amalgam of cloud computing and server based technologies for anonymous file sharing service.

Wang et al [1] observes user's motivation and intent for saving and sharing files on cloud using game theory. They observe that the process of file sharing is reduced and identify it as the existence of an infinitely repeated prisoner's dilemma (PD) game while the action of file sharing as an action of cooperation in the PD game. They incorporated win stay lose shift (WSLS) strategy and simulated it using tit-for-tat (TFT) strategy and found that WSLS is an efficient way to increase users' motivation to use file sharing systems.

Zhu et al [2] proposes a secure and practical attribute based encryption scheme without pairings (CP-ABE-WP) in cloud computing scenarios for a secure file sharing scheme based on attribute control. They identified that the current cloud based file sharing systems are unable to provide enough flexibility, fine-grained access control and access security. They have design a secure cloud file system by implementing CP-ABE-WP and they found that plaintext is secure in a selective ID model with acceptable performance and can satisfy the file sharing application in cloud computing.

Agrawal et al [3] explains that it is very difficult to apply additional security whenever a system is outsourcing its cloud. The writers propose encrypting files and generating symmetric keys using the General algorithm and re-encryption of files which ensures that the data will be safe even on untrusted clouds.

Lee et al [4] talks of how the relational SQL databases can be converted to a NoSQL database as they benefit the application via better performance and output then the traditional relational database while the execution of the ML algorithm on a Cloud Platform i.e Apache Spark & Hadoop.

Li-Wen Huang et al [5] proposed to use python based web application to create a board game by using Django web framework to create a GUI for the game and with the analysis and statistics generated while playing is used by ML to generate dead ends and new path on the board game to enhance playing experience.

Zouheir Daher et al [6] have done a comparative analysis of Amazon Simple Storage vs. Microsoft Azure Blob Storage. They found that Azure requires the same tier for all storage account objects, but Amazon S3 allows setting the access frequency-tier for each object. So from this they concluded that it is wise to use Azure Blob Storage for creating a dedicated CBS account to backup data without using it for primary storage. They stated that Azure and S3 are the best used where a lot of data needs to be stored but accessed infrequently with quick access when needed.

They identified that both Azure Blob Storage and Amazon S3 provide key management and encryption for data in & pre-flight but Azure is not offering post-flight encryption.

Pratiksha P. Nikam et al [7] explain that in the tests undertaken, the use of low-cost security add-on services provided no additional layer of security.

Furthermore, the services that have been set up to analyze the contents of their sites via FTP failed to identify the suspicious files.

Andrew Marshall et al [8] state that regardless of how carefully developers hide the key, they should not upload it or any keying information to Azure Storage as the secret key may be disclosed if any computer or storage services were compromised. There really is presently no support for Data Protection API (DPAPI)-like secret data persistence. If a program has to encrypt sensitive data at rest in Microsoft Azure Storage, it must do so offsite first, before transferring the encrypted payload to blob storage. This is simple to accomplish using an AES key created on a local computer

or somewhere else in the organisation. Microsoft advises using 256-bit AES keys for encryption.

Multi Clouds is something that is being explored to increase security, keeping files on different clouds will increase security exponentially. On the other hand, confidentiality can be maintained by using block ciphers for encryption. The block cipher will ensure data is not breached or read with the help of brute force. A block cipher operates on a chunk of data. Plain text is converted into cipher text using a cipher. The mapping is done in a random way such that only the key holder will know about it. One of the most well-known ciphers is the AES (Advanced Encryption Standard) block cipher. AES is very safe and is also certified by NIST. But even the AES cipher can be decrypted in some cases, if the blocks are not dependent on each other. Another downside of AES is the high computation use of hybrid cloud can solve this discrepancy. This has been proposed by Huang et al [15]. Their method had been tested in real-world network settings, such as Amazon EC2. Their technique achieves data privacy in a fraction of the time it takes the AES algorithm. When compared to the typical public-cloud-only strategy, their hybrid cloud approach takes 3 percent to 5 percent longer. [10] [11] [12][13]

Various tests and analyses have been conducted in the past to strengthen the security of the environment of cloud computing. We noticed that many researchers have implemented different security algorithms with databases on AWS EC2.

Paudyal et al and Eletriby et al [16] [14] have done an evaluation study comparing security algorithms like RC6, AES, DES, RSA and Blowfish. They came up with different findings.

III. WEB APPLICATION PARTS

A. Frontend part

To build the front end, developers use a combination of HTML (page layout and basic content), CSS (visual layout), and JavaScript (interactive websites). The same set of tools is used to create continuous web applications. The front end of the site is what you see and can connect to through the system. In our web service the front end development is done in flask using html files.

HTML

HTML (or Hypertext Markup Language) is a computer language designed to create websites that can be screened by anyone accessing the Internet. HTML is usually hired to create a web document. Defines things like news headlines or categories and enables embedding photos, video, and other media.

HTML is made up of a series of short codes called tags, usually a text file by a site builder. The text is then saved as an HTML file and viewed in a browser. The browser scans the file and translates the text into visual form, and in the best case, translates the page as planned by the developer.

HyperText is how we navigate the web by clicking links - certain texts that take you to other pages. Hyper means incompatible, allowing you to move to any other location, because there is no predetermined command to do so.

The quote determines the attributes that HTML tags use in the text within them. Tags tag it as a specific type of text.

As a language, it has the same code and syntax as any other language.

Text between `<html>` and `</html>` specifies a web page, while text between `<body>` and `</body>` determines visual content

HTML of our service is designed by keeping these philosophies in mind in order to provide the best user experience.

CSS

CSS (or Cascading Style Sheets) is a style sheet language. It is used to describe how HTML objects should be presented on a web page by the design, layout, and variety of different devices with different screen sizes. CSS performs well the layout of various web pages at a time.

CSS interacts with HTML objects, parts of a web page. To communicate with HTML, CSS uses selectors. Picker is part of the CSS code that describes which HTML will be taken by the CSS style. The declaration contains the properties and values used by the selector.

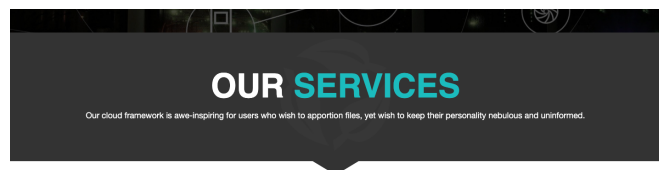
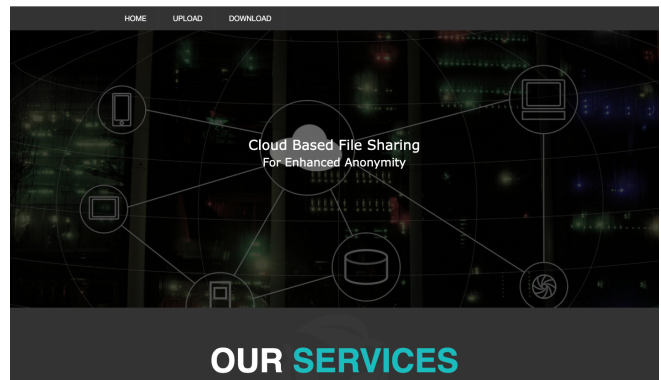
Properties define font size, color, and margins. Prices are the settings for these properties.

CSS frameworks. A CSS framework is a set of default CSS and HTML files. Increases the power of a leading developer in website design. In addition to assisting in the development of a responsive design, CSS frameworks also present unique and equal structures, saving engineers from compiling code from zero at all times. They are often considered to be a good choice to fit various platforms and screen sizes. With standard user elements, grid layouts, layouts, and many other features, CSS frameworks greatly accelerate the development of development work.

In order to optimize space so that users can number of files, we have incorporated css in HTML pages.

Screenshots of our frontend:

Home Page



UPLOAD

It is really easy to share your files with full SECURITY and ANONYMITY. After you complete an upload, a download link will be generated. You can share this link with anyone. You can share this link with anyone for them to download it.

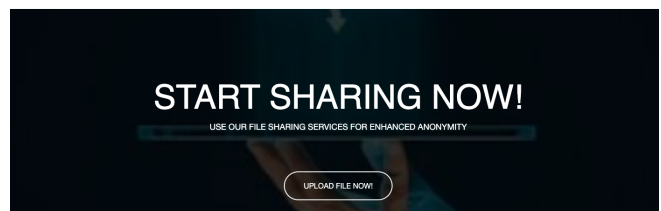
DOWNLOAD

It is really easy to share your files. After you complete an upload, a download link will be generated. You can share this link with anyone and when they visit this link they will see a download button and get your files easily.

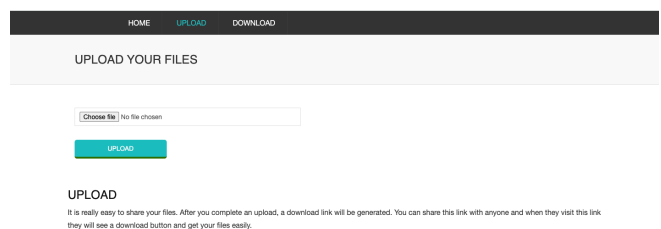


About Us

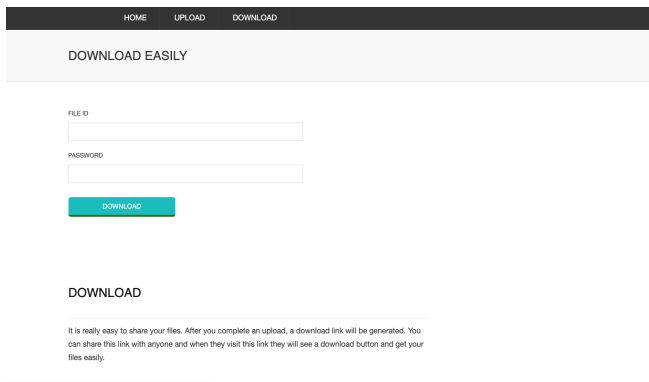
In today's world a lot of focus is given to security and very little to anonymity. There are a variety of contexts in which users have legitimate reasons to want to exchange information, while maintaining anonymity. In many file-sharing services, the users can be tracked. We have developed a file sharing service based on the Python programming language and cloud.



Upload Page



Download Page



B. Backend

Flask

A strong backend is a very important aspect of dynamic web-services like file sharing service. Backend provides users with access to the database of the system to access files using the internet.

In our service flask is being used as one of the backend components. Flask is a python based micro web framework enabling developers to get up and running with their backend quickly. We chose Flask for our work because it provides us fine grained control and is one of the latest technologies while this paper was written.

In our work Flask is responsible for uploading and downloading of files by the users which are then stored on cloud. While uploading a file users have to provide file id and 2 passwords for encryption and privacy of the file being shared. Flask is responsible for passing file id, uploaded file and passwords to the database.

Whilst downloading a file Flask matches the passwords given by the user for a given file id and if the password is correct it gets the requested file from the databases so that it can be downloaded by the user.

C. Database

The database in this project is a NoSQL database, i.e MongoDB, which is used to store the details when the service is used to upload a file. the database service is hosted on a cloud platform itself which is also provided by the creators of mongodb called “MONGODB ATLAS” which is a multi-cloud platform allows to host the service with major cloud providers but also takes care of security and analytics, thus allows us to scale up or scale down as required and support for applications with edge-to-cloud sync and fully managed backend services

To create the service we simply went to their domain[17] >> create an account >> create a project >> select a cloud provider and the size of instance required and get started. Next is to generate the user token along with the password to access the service in our flask web application.

Mongodb is a document based database meaning that the data is stored in “BSON” which stands for Binary JSON and thus to fulfil the limitations of SQL databases mongodb also allows to store data in various different formats.

To Interact with the python backend thus python package pymongo is required which can be installed with this command on a cmd/terminal

“python -m pip install pymongo”

and we also require dnspython package as we are using mongodb srv URI and can be installed with this command on the cmd/terminal

“python -m pip install dnspython”

next step required it to connect the database to our backend by adding following lines of code

“# Provide the mongodb atlas url to connect python to mongodb using pymongo

CONNECTION_STRING="mongodb+srv://<username>:<password>@<cluster-name>.mongodb.net/filesaring"

Create a connection using MongoClient. You can import MongoClient or use pymongo.MongoClient

```
from pymongo import MongoClient

client = MongoClient(CONNECTION_STRING)
```

Whenever a file upload event take place the backend processes the file and sends the data to a function to create a collection of data in form of python dictionary which then is converted into BSON by the pymongo package and is stored on the database service

<dict code here form db_upload >

and in case of a file download event the the file_id is collected from the user and is passed to the function dedicated to retrieve data related to the file asked for

D. Azure Cloud Storage

A storage account is just a container that holds a collection of services. When data services are integrated into a storage account, the user can manage them as a group. The account settings that you specify when you create it, or that you update afterward, apply to all of your devices.

Microsoft Azure Blobs

Blob stands for Binary Large Object, which includes objects such as images and multimedia files.

Blobs are used to store all kinds of data that is meant to be shared with others, such as files, images, and videos. They can also hold raw data from scientific research or other large amounts of data.

Blob Storage in Azure is designed to store large amounts of unstructured data. There are three kinds of resources available in blob storage: storage, container and archive.

The time it takes to analyse big data sets is reduced when they are already in the cloud, such as with metadata and reference data.

Blob storage <https://<storage-account>.blob.core.windows.net>

To manage blobs with Azure Storage, first configure your Azure service and check to see if it's up and running.

Create the project i.e. python application named blob-quickstart-v12.

Type this in the console :

```
mkdir blob-quickstart-v12
```

then, cd blob-quickstart-v12

```
mkdir data
```

Then, install the Azure Blob Storage client library for Python package

In the console enter : pip install azure-storage-blob

Next, Set up the app framework

```
import os, uuid
```

```
from azure.storage.blob import BlobServiceClient,
BlobClient, ContainerClient, __version__
```

try:

```
print("Azure Blob Storage v" + __version__ + " - Python
quickstart sample")
```

```
# Quick start code goes here
```

except Exception as ex:

```
print('Exception:')

```

```
print(ex)
```

Save the file in .py format in the directory.

Now, copy your credentials from the Azure portal

and configure your storage connection string.

```
setx AZURE_STORAGE_CONNECTION_STRING
"<yourconnectionstring>"
```

Now you can perform different tasks with the Azure Blob Storage client library for Python like:

Getting the connection string then, creating Blob storage containers, Containers are similar to folders and store blobs. The images and files container is where the app uploads images and files. We create containers using the New-AzStorageContainer command.

Upload blobs to a container i.e. Uploading the local text file to the blob by calling the upload_blob method.

The blobs can be listed by the calling the list_blobs method.

Blobs can be downloaded by calling the download_blob method.

To summarise, this programme creates and uploads a test file to Azure Blob Storage from your local folder. The example then displays a list of the blobs in the container before downloading the file under a new name.

E. Cloud Deployment

EC2

SaaS (software as a service), PaaS (platform as a service), and IaaS (infrastructure as a service) solutions that may be accessed on demand by end users or customers are referred to as cloud deployment. Before user provisioning can take place, all of the necessary installation and configuration processes must be completed in the cloud. Some cloud deployment models are public, private, hybrid, and community models. To build out a project on an Amazon Elastic Compute Cloud (EC2) instance, there are many sizing options available, as well as multiple scalability methods for expanding storage and computation capabilities. With choices in processor, storage, networking, operating system, and buying model, Amazon EC2 is the most thorough and complete computational platform available. It also has the fastest processors in the cloud which are the only cloud with ethernet networking speed of 400 Gbps.

To run your own flask application on the newly generated instance, first create your EC2 instance and check to see if it's up and running.

Next Deploy to Production, just use app.run function

Now SSH into the virtual machine,

Enter in your bash and locate the .pem file previously downloaded. Type `chmod 600 ./<PEM_NAME>.pem` to restrict read and write permission to the private key, so only the user/owner can read and write, but no one can execute. After that, you can log into the virtual machine typing

```
$ ssh -i path/<PEM-NAME>.pem ubuntu@<IP-ADDRESS>
```

For the Virtual machine setup, proceed to install the dependencies.

```
$ sudo apt update
```

```
$ sudo apt install python3 python3-pip tmux http
```

Now you can create a new folder that will contain your application.

```
$ mkdir deployed_app
```

Then proceed to install the dependencies and then at last deploy the application.

These are the reason which has caused us to believe to create a anonymous file sharing system at an amature level our backend stack is the best choice the implementation of our service indicated the same.

IV. CONCLUSION

This paper presents a file sharing system that promotes user anonymity and quick file sharing. Through literature survey one can observe that not much research has been done on the presented topic. Nevertheless, this paper fills the chasm that exists between cloud technologies and traditional file

sharing services. Thereby leveraging the advantages of latest technologies like Flask, Mongo DB, AWS and Azure, the paper has presented an implementation of an anonymous file sharing system.

IV. REFERENCES

- [1] Yilei Wang¹, Willy Susilo², Tao Li³, Qiuliang Xu⁴ "File sharing in cloud computing using win stay lose shift strategy" Department of Computer Science and Technology, Shandong University, China Copyright © 2015 Inderscience Enterprises Ltd.
- [2] Shuaishuai Zhu and Xiaoyuan Yang "Protecting data in cloud environments with attribute-based encryption". pp 91-97 .Electronics Department, Engineering University of the Armed Police Force, Xi'an 710086, China Copyright © 2015 Inderscience Enterprises Ltd.
- [3] Neha Agarwal, Ajay Rana and Jai Prakash Pandey "An efficient and optimised approach for secured file sharing in cloud computing" Amity University, Sec 125, Noida UP, India Copyright © 2021 Inderscience Enterprises Ltd.
- [4] C. H. Lee and Z. W. Shih, "A Comparison of NoSQL and SQL Databases over the Hadoop and Spark Cloud Platforms Using Machine Learning Algorithms," 2018 IEEE Int. Conf. Consum. Electron. ICCE-TW 2018, vol. 2, no. c, pp. 1–2, 2018, doi: 10.1109/ICCE-C hina.2018.8448621.
- [5] L. -W. Huang, P. -H. Cheng and L. -W. Chen, "Web-based Board Game for Learning Python," 2021 IEEE World Conference on Engineering Education (EDUNINE), 2021, pp. 1-6, doi: 10.1109/EDUNINE51952.2021.9429144.
- [6] Daher, Z. & Hajjdiab, Hassan. (2018). Cloud storage comparative analysis amazon simple storage vs. microsoft azure blob storage. International Journal of Machine Learning and Computing. 8. 85-89. 10.18178/ijmlc.2018.8.1.668.
- [7] Pratiksha P. Nikam, Ranjeetsingh S. Suryawanshi, "Microsoft Windows Azure: Developing Applications for Highly Available Storage of Cloud Service", International Journal of Science and Research (IJSR), https://www.ijsr.net/get_abstract.php?paper_id=NOV15186 4, Volume 4 Issue 12, December 2015, 662 - 665
- [8] Marshall , A., Howard, M., Bugher, G., & Harden , B. (2010, June). "Security Best Practices For Developing Windows Azure". Security Best Practices For Developing Windows Azure. Retrieved October 23, 2021, from http://download.microsoft.com/documents/uk/enterprise/88_security_best_practices_for_developing_windows_azure_applikat.pdf.
- [9] Quickstart: Manage blobs with Python v12 SDK [Online]. Available:
- [10] Ahad Niknia, Miguel Correia, Jaber Karimpour, Secure cloud-of-clouds storage with space-efficient secret sharing, Journal of Information Security and Applications, Volume 59, 2021, 102826, ISSN 2214-2126, <https://doi.org/10.1016/j.jisa.2021.102826>.
- [11] Timothy J. Shimeall, Jonathan M. Spring, Chapter 8 - Resistance Strategies: Symmetric Encryption, Editor(s): Timothy J. Shimeall, Jonathan M. Spring, Introduction to Information Security, Syngress, 2014, Pages 155-186, ISBN 9781597499699, <https://doi.org/10.1016/B978-1-59749-969-9.00008-0>.
- [12] Tom St Denis, Simon Johnson, Chapter 4 - Advanced Encryption Standard,

Editor(s): Tom St Denis, Simon Johnson, Cryptography for Developers, Syngress, 2007, Pages 139-202, ISBN 9781597491044,

<https://doi.org/10.1016/B978-159749104-4/50007-8>.

(<https://www.sciencedirect.com/science/article/pii/B9781597491044500078>)

[13] Fursan Thabit, Sharaf Alhomdy, Sudhir Jagtap,

Security analysis and performance evaluation of a new lightweight cryptographic algorithm for cloud computing,

Global Transitions Proceedings, Volume 2, Issue 1, 2021,

Pages 100-110, ISSN 2666-285X,

<https://doi.org/10.1016/j.gltp.2021.01.014>.

[14] Eletriby, Sherif & Meslhy, Eman & Abd elkader,

Hatem. (2012). Modern Encryption Techniques for Cloud Computing Randomness and Performance Testing.

10.13140/2.1.4685.8880.

[15] Huang, Xueli & Du, Xiaojiang. (2013). Efficiently secure data privacy on hybrid cloud. IEEE International Conference on Communications. 1936-1940.

10.1109/ICC.2013.6654806.

[16] Paudyal, Ramesh & Shakya, Subarna. (2021). Secure Data Mobility in Cloud Computing for e-Governance Application. Journal of Engineering Technology and Planning. 2. 1-14. 10.3126/joetp.v2i1.39203.

[17] <https://www.mongodb.com/cloud/atlas>